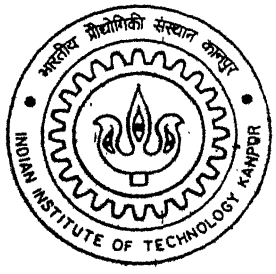
LIFE ASSESSMENT OF THERMAL POWER PLANT COMPONENT USING ARTIFICIAL NEURAL NETWORKS

By

Ajay Kumar Singh Tomar



NUCLEAR ENGINEERING AND TECHNOLOGY PROGRAMME

Indian Institute of Technology Kanpur

OCTOBER, 2001

LIFE ASSESSMENT OF THERMAL POWER PLANT COMPONENT USING ATRIFICIAL NEURAL NETWORKS

A Thesis Submitted

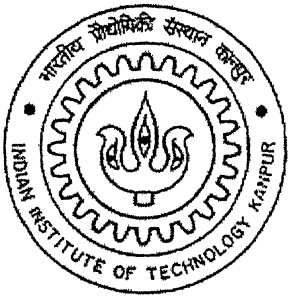
In Partial Fulfillment of the Requirements

For the Degree of

Master of Technology

By

Ajay Kumar Singh Tomar



Nuclear Engineering and Technology Programme

Indian Institute Of Technology Kanpur

October 2001

CERTIFICATE

It is certified that the work contained in the thesis entitled ***Life Assessment of Thermal Power Plant Components using Artificial Neural Networks***, by ***Ajay Kumar Singh Tomar***, has been carried out under our supervision and that this work has not been submitted elsewhere for a degree.

Dr. K. Deb
Professor
Department of Mech. Engg.
Indian Institute of Technology
Kanpur 208016

Dr. N. N. Kishore
Professor
Department of Mech. Engg.
Indian Institute of Technology
Kanpur 208016

October, 2001

ACKNOWLEDGEMENT

First of al I would like to thank Prof. K. Deb, Prof. N. N. Kishore and Dr. B. K, Dutta for all the support, for never ending stream of ideas and for always taking the time for discussions. Without them this thesis had never been written.

Another person I owe thanks is Prof. Prabhat Munshi for always being supportive to me. His help, academically and otherwise, is gratefully acknowledged.

Thanks to my friends Ritesh, Mukesh, Dr. A. M. Quraishi, Sanjeev and Suneet for all the help and cooperation I got from them.

Abstract

In the design of thermal power plant components, safety is the prime consideration. Generally components are designed as per the guidelines of ASME Boiler and Pressure Vessel Codes. In the design, the life of the plant is estimated based on assumed process transients. Because of the conservatism in the design actual life of the components is more than estimated life. Recently the issue of remnant life of the components has attracted considerable attention in the power industry.

Aging of the components is one of the prime considerations for safe operation of the plant and fatigue is the common aging mechanism. Hence to estimate the remnant life of the power plant components, on-line monitoring of fatigue degradation is necessary.

In the present work feasibility study of using Artificial Neural Networks in conventional Finite Element Based on-line fatigue monitoring system is done. Feed Forward Error Back-propagation neural network was trained and tested for the data obtained by FEM code, which converts transient in steam temperature to thermal stress response of the component. Effect of over training on the performance of the network is also studied.

Nomenclature

α Thermal coefficient of expansion.

A Area of the element.

[B] Matrix relating displacements with strains.

[D] Constitutive matrix.

ε Strain.

f Distributed force per unit volume.

[K] Global stiffness matrix.

n_i Direction cosine of normal to the surface.

n_n Number of stress cycles of stress level σ_n, already being seen by the component.

[N] Matrix of shape functions.

N_n Fatigue life of the component at stress level σn.

η Learnung rate constanat.

Π Potential energy.

P_b Body force.

P_S Surface force.

σ Stress

$\bar{t}$ Surface Traction.

t Transient build up time.

ΔT Change in steam temperature before operator takes corrective measures.

u Displacement.

U Strain energy.

ν Poison's ratio.

$W_{j,i}$ Weight connecting i^{th} neuron to j^{th} neuron.

x Input vector.

List of Tables

List of Figures

Contents

Chapter 1

Introduction

Computer simulation of the behavior of complex systems and components, whose dynamic representation typically requires the solution of many equations and extensive use of closure relations, has become very important in modern design. This way of proceeding is used particularly in the nuclear industry where safety rules are rather rigid and impose to closely examine any possible situation and mode of operation of the system.

The traditional approach to dynamic simulation is based on the realization of a simulation model in three steps: understanding the underlying physics, defining the mathematical model and developing a fast and reliable code. This is a challenging process that needs to be carried out with care. For example, the underlying physics may contain significant non-linearities that render the associated mathematical model quite complicated. On the other hand, the simplifications and approximations, which are necessarily performed, may lead to the introduction of uncertainties. Moreover, in the successive phase of implementation of the physical-mathematical model, issues of stability and convergence can complicate the discretization of the differential systems. Still, the main problem of numerical simulations is that if the equations of the model are strongly coupled and the characteristic times of the involved phenomena differ substantantially, the traditional methods of solution may involve long, iterative processes which result in simulation times significantly larger than the real time of the transient.

Artificial Neural Networks (ANN) can be used in this context to somewhat alleviate this problem. Their parallel and multi-parametric character, and their great speed of computing have permitted to assert themselves as a powerful computational tool in many fields of research and application, especially where traditional techniques involve long elaboration times or when it is particularly difficult to define the underlying physical-mathematical model [10].

The advantages that result from the ANN as a simulation tool are found not only

necessary, in principle, to completely unravel the physics that governs the plant, since ANN is capable of constructing its own internal representation of the system input/output relations on the basis of the training set.

Moreover, it is not necessary to build and solve a system of partial differential equations but it is sufficient to have some data covering all the possible situations of operation, for the ANN training. The generation of training set can be done either by a numerical code or from actual plant operation data. What is important is that the training data cover all the working space that we are going to simulate.

The objective of the present work is to study the feasibility of neural network for the on-line computation of transient temperature and thermal stresses in a thick pipe. The ANN needs training data to train the network which is obtained either by experimentation or by numerical simulation. In the present work Finite Element Method (FEM) has been used as a numerical technique to obtain the temperature-stress response of the structure, which will be used to train and test the network.

1.1 Problem Formulation

The Fig. 1.1 shows the schematic of a thick pipe for which temperature and stress response of pipe material is predicted using adaptive time delay neural network. Steam at constant pressure and high temperature flows through it with constant flow rate. Transient Stress and Temperature response of the pipe material is to be simulated for step change in temperature. Following steps are to be carried out in the present project:

Step 1

Generate training and testing data using FEM, which converts plant transients to temperature and stress response of the pipe material.

Step 2

Construct and train the network with MATLAB's NN TOOLBOX for above data and optimize network topology.

Step 3

Test the above network for new input data not seen during training.

Step 4

Predict the remaining life of the component to avoid disaster.

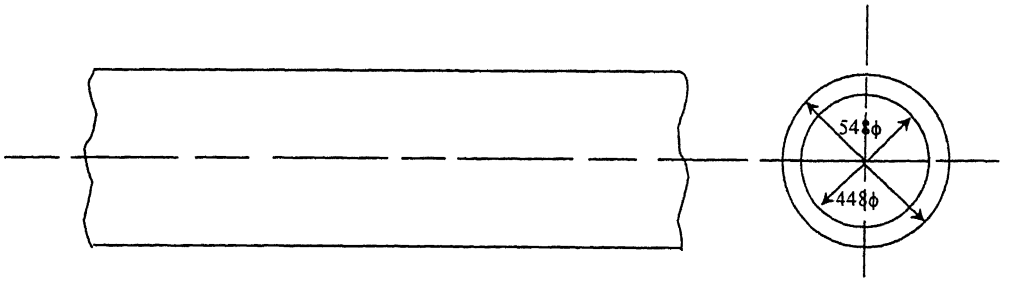


Fig. 1.1.1 Schematic of a thick pipe for which stress response of pipe material is predicted

1.2 Thesis Layout

- Chapter 2 describes detailed problem description and basics of FEM used to solve solid mechanics problems.
- Chapter 3 gives details neural networks and training and testing data generation using FEM.
- Chapter 4 gives simulation results.
- Chapter 5 summarizes the conclusion and suggests the scope for future work.

Chapter 2

Problem Description

Main working fluid of thermal power plants is steam, which flows through thick pipes. Thermal power plant operates with several plant parameters like temperature, pressure and flow rate of the steam and plant is supposed to run at their constant values. But in plant operation it's very difficult to have a control over them. Malfunctioning of any plant component results in the fluctuation of these plant parameters. When operator senses fluctuation in plant parameter, he takes some corrective action to bring it to their normal plant operation values. In the present work the fluctuation in the steam temperature is taken into consideration. Fluctuation in the steam temperature will result in the cyclic fluctuation in the thermal stress in the pipe, which in turn will result into the fatigue of the material. Fatigue is the main aging i.e. degradation of material with time, mechanism and hence, common cause for most of the failures in power plant components. This effect is the most dominant when the plant approaches the end of the designed life. Hence to estimate the remnant life of the power plant components, on-line monitoring of fatigue life is necessary [6]. Following are the common methodologies used for fatigue monitoring:

1. Analytical solution and/or empirical equations as per the standard codes.
2. Green Function Technique (GFT).
3. Finite Element Methods.

On-Line Fatigue Monitoring System based on FEM has been successfully developed and commissioned by Reactor Safety Division, BARC, Mumbai. That converts plant transients to Temperature-Stress response of the structure using FEM. Computation with FEM for On-Line Monitoring of fatigue is computationally expensive and sometimes the

time taken by FEM module to convert plant transients to temperature-stress response of the structure is more than the duration of the transient itself [7]. To study the feasibility of replacing the FEM Module with Artificial Neural Network Module is the main objective of the present work. Advantages of such a scheme are as follows:

1. ANN can be trained off-line so on-line computational time will be very less in comparison with that of FEM.
2. For better accuracy FEM needs very fine mesh that is analyzed at every node. This process creates large matrices and requires large memory space of computing machine. Also even a single data corruption can cause trouble to the process and ultimately accuracy will suffer. While what ANN needs is a storage of weights that is created during training phase.

2.1 Analysis Procedure

In the present project On-Line monitoring of a thick pipe as described in Fig 1.1.1 through which steam flows at high temperature is done at two critical points viz. inner radius and outer radius. For on-line monitoring quantities of interest are: maximum stress, minimum stress, their difference and the time instant when these occur at both the critical points.

Problem can be divided in three parts: Generation of training and testing data for ANN by FEM for various boundary conditions (input to the NN are difference in temperature of the steam from normal operating temperature and duration of the transient while output of the NN are maximum stress, minimum stress, their difference and the time instant when these occur), Training of the Neural Network (code available in MATLAB's NN TOOL BOX) and Testing the Network for the input values not seen in training. In the next section FEM methodology is reviewed and in next chapter it's use in training and testing data generation is described.

2.2 Finite Element Method

Finite Element Method (FEM) is a numerical technique to obtain numerical solution to a wide variety of engineering problems. Though simple idealized problems have exact solutions, in general, the real engineering problems are too complex to permit an exact solution. They demand quantitative predictions. They have physical parameters that need constant study and are not amenable for exact solution. In such real problems 'numerical approximation is the only resource to obtain quantitative predictions. At present several numerical methods are available, namely, finite difference, finite elements, finite volumes, Power series, spectral methods, boundary solutions etc. FEM is extensively used in various fields such as structural mechanics, solid mechanics, thermodynamics, electromagnatics etc. The applications include the design of automobiles, airframes, high-rise buildings, spacecrafts, heat engines, electric motors, bearings etc.

The FEM may be described as a numerical method in which the unknown function in the problem domain is approximated by piecewise defined functions. That is, the domain is divided into several sub-domains of simple shapes and function is approximated separately in each sub-domain by simple polynomial functions. The governing equations are derived in a overall sense as a summation of elemental equations. The sub-domains are called 'elements' and they are interconnected at points known as 'nodes'.

FEM has its maximum application in structural mechanics and the solid mechanics. Solid mechanics deals with the general formulation of deformable bodies subjected to forces and displacements, stresses and strains. In following sections basic solid mechanics concepts like equilibrium equations, compatibility condition, constitutive relations and general FEM formulation for solid mechanics problems are reviewed.

2.3 Basic Solid Mechanics Concepts

A three-dimensional body occupying a volume V and bounded by surface S is shown in Fig. 2.1. Points in the body are located by x, y, z coordinates. The boundary is constrained on some region, where displacements are specified. On part of the boundary, distributed force per unit area $\bar{t}$, also called traction, is applied. Under the force body

deforms. The deformation of a point P (x,y,z) is given by three components of its displacements:

$$\mathbf{u} = [u,v,w]^T \tag{2.1}$$

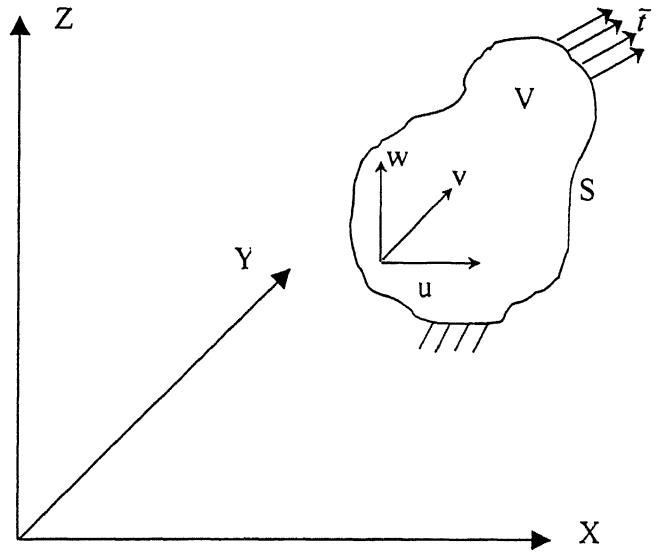


Fig. 2.1 General three-dimensional body

The distributed force per unit volume, for example, the weight per unit volume, is the vector f given by:

$$f = \left[\overline{X}, \overline{Y}, \overline{Z}\right]^T \tag{2.2}$$

The surface traction $\bar{t}$ may be given by its components at points on the surface as:

$$\bar{t} = [t_x, t_y, t_z]^T \tag{2.3}$$

Stress σ acting at a point can be represented by a 3×3 symmetric matrix:

$$\sigma = [\sigma_x, \tau_{xy}, \tau_{xz};\ \ \tau_{xy}, \sigma_y, \tau_{yz};\ \ \tau_{xz}, \tau_{yz}, \sigma_z,] \tag{2.4}$$

Consideration of force balance in x-, y- and z- directions in bulk V give equilibrium equations in x-, y- and z- direction respectively.

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} + \overline{X} = 0 \tag{2.5}$$

$$\frac{\partial \tau_{xy}}{\partial x}+\frac{\partial \sigma_y}{\partial y}+\frac{\partial \tau_{yz}}{\partial z}+\bar{Y}=0 \tag{2.6}$$

$$\frac{\partial \tau_{xz}}{\partial x}+\frac{\partial \tau_{yz}}{\partial y}+\frac{\partial \sigma_z}{\partial z}+\bar{Z}=0 \tag{2.7}$$

Equilibrium equations on surface S give traction boundary conditions:

$$\sigma_x n_x+\tau_{xy} n_y+\tau_{xz} n_z=\bar{t}_x \tag{2.8}$$

$$\tau_{xy} n_x+\sigma_y n_y+\tau_{yz} n_z=\bar{t}_y \tag{2.9}$$

$$\tau_{xz} n_x+\tau_{yz} n_y+\sigma_z n_z=\bar{t}_z \tag{2.10}$$

where n_x, n_y and n_z are directional cosines of the external normal on the traction boundary Γ_q.

Strain-Displacement Relations

Strain ε at a point, say P, can be represented by a 3×3 matrix as:

$$\varepsilon = [\varepsilon_x, \gamma_{xy}, \gamma_{xz};\ \ \gamma_{xy}, \varepsilon_y, \gamma_{yz};\ \ \gamma_{xz}, \gamma_{yz},\ \varepsilon_z] \tag{2.11}$$

For small deformations strain ε at P can be written in terms of displacement u at P as:

$$\varepsilon = [\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}, \frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}; \frac{\partial u}{\partial y}+\frac{\partial y}{\partial x}, \frac{\partial y}{\partial y}, \frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}; \frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}+\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}, \frac{\partial w}{\partial z}] \tag{2.12}$$

Compatibility Condition

These are the additional conditions based on the fact that the displacements are continuous and single valued at every point in the body. From this property compatibility condition can be written as:

$$\frac{\partial^2 \varepsilon_y}{\partial x^2} + \frac{\partial^2 \varepsilon_x}{\partial y^2} = \frac{\partial^2 \gamma_{xy}}{\partial x \partial y} \tag{2.13}$$

Constitutive Relations

For linear elastic materials, the stress-strain relations come from generalized Hooke's law. For isotropic materials, the two material properties are Young's Modulus of Elasticity and Poisson's ratio. For linear elastic isotropic material stress-strain relation is given as:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{yz} \\ \tau_{zx} \\ \tau_{xy} \end{Bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{yz} \\ \gamma_{zx} \\ \gamma_{xy} \end{Bmatrix} \tag{2.14}$$

Or,

$$\{\sigma\} = [D]\{\varepsilon\}$$

where $[D]$ is constitutive matrix.

2.4 Basic FEM Formulation for Solid Mechanics Problems

There are two commonly used FEM formulations for solid mechanics problems-*Galerkin Weighted Residual Formulation* and *Variational approach.* In present project variational approach has been used hence, will be reviewed here.

2.4.1 Variational Approach

In structural mechanics there exist several variational methods such as minimum potential energy, maximum complementary energy, stationary principle of Hellinger-

Reisner etc. The principle of minimum potential energy is a very popular variational principle. That is, out of all displacement fields satisfying displacement boundary conditions, the displacement field, that minimizes the potential energy is the one that satisfies the governing equilibrium equations. The potential energy Π involves the internal strain energy of the body and the potential due to external forces. That is,

$$\Pi = U - W \tag{2.15}$$

where U is the total strain energy of the body and $-W$ is the potential due to external forces. Since strain energy density at a point is given by

$$dU = \frac{1}{2}(\sigma_x \varepsilon_x + \sigma_y \varepsilon_y + \sigma_z \varepsilon_z + \tau_{yz}\gamma_{yz} + \tau_{zx}\gamma_{zx} + \tau_{xy}\gamma_{xy}) \tag{2.16}$$

Potential due to external forces is given by

$$W = \int_v (\bar{X}u + \bar{Y}v + \bar{Z}w)dv + \int_{\Gamma_q} (\bar{t}_x u + \bar{t}_y v + \bar{t}_z w)d\Gamma \tag{2.17}$$

Hence, Potential Energy of the body can be given as:

$$\begin{aligned} \Pi = & \frac{1}{2}\int_v (\sigma_x \varepsilon_x + \sigma_y \varepsilon_y + \sigma_z \varepsilon_z + \tau_{yz}\gamma_{yz} + \tau_{zx}\gamma_{zx} + \tau_{xy}\gamma_{xy})dV \\ & - \int_v (\bar{X}u + \bar{Y}v + \bar{Z}w)dv - \int_{\Gamma_q} (\bar{t}_x u + \bar{t}_y v + \bar{t}_z w)d\Gamma \end{aligned} \tag{2.18}$$

Where integrations are carried over the whole volume of the body and on the surface where tractions are prescribed. This can expressed concisely using matrix notation as,

$$\Pi = \frac{1}{2}\int_v \{\varepsilon\}^T\{\sigma\}dv - \int_v \{\bar{X}\}^T\{u\}dv - \int_{\Gamma_q} \{\bar{t}\}^T\{u\}d\Gamma \tag{2.19}$$

In finite element analysis the P.E. of the whole body is expressed as the summation of PE over all the elements as

$$\Pi = \sum_{e=1}^{E} \Pi^{(e)} \tag{2.20}$$

Potential Energy for one element can be calculated as

$$\Pi^{(e)} = \frac{1}{2}\int_{v^{(e)}} \{\varepsilon\}^T\{\sigma\}dv - \int_{v^{(e)}} \{\bar{X}\}^T\{u\}dv - \int_{\Gamma_q} \{\bar{t}\}^T\{u\}d\Gamma \tag{2.21}$$

or,

$$\Pi^{(e)} = \frac{1}{2}\int_{v^{(e)}} \{\varepsilon\}^T [D]\{\varepsilon\} - \int_{v^{(e)}} \{\bar{X}\}^T \{u\} dv - \int_{\Gamma_q} \{\bar{t}\}^T \{u\} d\Gamma \tag{2.22}$$

where [D] is constitutive matrix which related stress tensor to strain tensor as given in equation (2.10 a)

Above can be written in terms of displacements as

$$\Pi^{(e)} = \int_{v^{(e)}} \{u\}^T [B]^T \{D\}[B]\{u\} dv - \int_{v^{(e)}} \{u\}^{(e)^T} [N]^T \{X\} dv - \int_{\Gamma_q} \{u\}^{(e)^T} [N]^T \tilde{t}\, ds \tag{2.23}$$

where $[B]$ relates displacements with strain and $[N]$ is the matrix of shape functions. Element displacement vector is related with nodal displacement by shape functions as

$$\begin{Bmatrix} u \\ v \\ w \end{Bmatrix} = \sum_j \begin{bmatrix} N_j & 0 & 0 \\ 0 & N_j & 0 \\ 0 & 0 & N_j \end{bmatrix} \begin{Bmatrix} u_j \\ v_j \\ w_j \end{Bmatrix} \tag{2.24}$$

and $v^{(e)}$ is the domain of the element. The expression $\int_{v^{(e)}} [B]^T \{D\}[B] dv$ is known as *elemental stiffness matrix* and is represented as $[K]^{(e)}$, and $\int_{\Gamma_q} \{u\}^{(e)^T} [N]^T \tilde{t}\, ds$ is known as *elemental force vector* $P_S^{(e)}$. Hence,

$$\Pi = \sum_e \frac{1}{2} \{u\}^{(e)^T} \left[K^{(e)}\right] \{u\}^{(e)} - \sum_e \{u\}^{(e)^T} \{P\}_b^{(e)} - \sum_e \{u\}^{(e)^T} \{P\}_S^e \tag{2.25}$$

above elemental matrices can be summed to Global Matrices by the process of assembly

$$\Pi = \frac{1}{2}\{u\}^T [K]\{u\} - \{u\}^T \{P_b\} - \{u\}^T \{P_S\} \tag{2.26}$$

where $[K]$ is Global stiffness Matrix i.e. matrix obtained by assembling elemental stiffness matrices, and

$\{u\}$ is global nodal displacement vector.

As per minimum PE principle, Π has to be minimized w.r.t. $\{u\}$.

Hence, considering

$$\frac{\partial \Pi}{\partial \{u\}} = \{0\} \tag{2.27}$$

gives

$$[K]\{u\} - \{P_b\} - \{P_s\} = \{0\} \tag{2.28}$$

Or.

$$[K]\{u\} = \{P\} \tag{2.29}$$

$\{P\}$ includes concentrated load, if there are any.

In present problem geometry and load both are axisymmetric, so problem can be solved with axisymmetric methods of FEM. Element selected for the analysis is triangular ring type element. Next subsection reviews the FEM analysis of axisymmetric solids with triangular ring type element.

2.5 FEM Analysis for Axisymmetric Solids

There are many practical applications where the bodies are symmetric about one axis of symmetry such as, cooling tower, engine cylinder etc. Revolving a plane cross-section about the axis of symmetry can generate such bodies. If the forces acting on the solid are also axisymmetric then the solution i.e. displacement, strains, stresses etc.) will also be axisymmetric. In these cases, geometry, material, loading, boundary conditions are all independent of angular coordinate θ leading to tangential displacement, v=0, and the radial and axial displacements u and w are functions of r and z only. Though these problems are three-dimensional, it is enough to analyze the radial cross-section (rz-plane) only.

$$u = u(r,z), \qquad v = 0, \qquad w = w(r,z) \tag{2.30}$$

Therefore, for axisymmetric problems the strain tensor in polar cylindrical coordinates can be written as:

$$\{\varepsilon\}=\begin{bmatrix} \varepsilon r & \gamma_{r\theta} & \gamma_{rz} \\ \gamma_{r\theta} & \varepsilon\theta & \gamma_{\theta z} \\ \gamma_{rz} & \gamma_{\theta z} & \varepsilon z \end{bmatrix}=\begin{bmatrix} \dfrac{\partial u}{\partial r} & 0 & \dfrac{\partial u}{\partial z}+\dfrac{\partial w}{\partial r} \\ 0 & \dfrac{u}{r} & 0 \\ \dfrac{\partial u}{\partial z}+\dfrac{\partial w}{\partial r} & 0 & \dfrac{\partial w}{\partial z} \end{bmatrix} \tag{2.31}$$

Its obvious from Hooke's Law that there will be only four non zero stress components and $\tau_{r\theta}=\tau_{\theta z}=0$. Hence non-zero stress components are related to non-zero strain components as:

$$\begin{Bmatrix} \sigma_r \\ \sigma_\theta \\ \sigma_z \\ \tau_{rz} \end{Bmatrix}=\frac{E}{(1+\nu)(1-2\nu)}\begin{bmatrix} 1-\nu & \nu & \nu & 0 \\ \nu & 1-\nu & \nu & 0 \\ \nu & \nu & 1-\nu & 0 \\ 0 & 0 & 0 & \dfrac{1-2\nu}{2} \end{bmatrix}\begin{Bmatrix} \varepsilon_r \\ \varepsilon_\theta \\ \varepsilon_z \\ \gamma_{rz} \end{Bmatrix} \tag{2.32}$$

In the FE formulation of axisymmetric problems, the element is also axisymmetric ring type that is generated by revolving a plane element in rz-plane about z-axis. If the axisymmetric element has triangular cross-section in rz-plane and i,j and k represent the nodes of the triangle. Then displacement components can be written in terms of nodal displacements as

$$\begin{Bmatrix} u(r,z) \\ w(r,z) \end{Bmatrix}=\begin{bmatrix} N_i & 0 & N_j & 0 & N_k & 0 \\ 0 & N_i & 0 & N_j & 0 & N_k \end{bmatrix}\begin{Bmatrix} u_i \\ w_i \\ u_j \\ w_j \\ u_{ki} \\ w_k \end{Bmatrix} \tag{2.33}$$

where N_i, N_j and N_k are shape functions. And strain components are related to the nodal displacements by the relation:

$$\begin{Bmatrix} \varepsilon_r \\ \varepsilon_\theta \\ \varepsilon_\theta \\ \gamma_{rz} \end{Bmatrix} = \begin{bmatrix} b_i & 0 & b_j & 0 & b_k & 0 \\ \frac{N_i}{r} & 0 & \frac{N_j}{r} & 0 & \frac{N_k}{r} & 0 \\ 0 & c_i & 0 & c_j & 0 & c_c \\ c_i & b_i & c_j & b_j & c_c & b_k \end{bmatrix} \begin{Bmatrix} u_i \\ w_i \\ u_j \\ w_j \\ u_k \\ w_k \end{Bmatrix} = [B]\{u\}^{(e)} \qquad (2.34)$$

where

$$a_i = \frac{(x_j y_k - x_k y_j)}{2A} \qquad b_i = \frac{(y_j - y_k)}{2A} \qquad c_i = \frac{(x_k - x_j)}{2A}$$

$$a_j = \frac{(x_k y_i - x_i y_k)}{2A} \qquad b_j = \frac{(y_k - y_i)}{2A} \qquad c_j = \frac{(x_i - x_k)}{2A}$$

$$a_k = \frac{(x_i y_j - x_j y_i)}{2A} \qquad b_k = \frac{(y_i - y_j)}{2A} \qquad c_k = \frac{(x_j - x_i)}{2A}$$

here (x_i, y_i), (x_j, y_j) and (x_k, y_k) are the coordinates of nodes i, j and k respectively and A is the area of the triangular section of the ring type axisymmetric element in rz-plane.

Now elemental stiffness matrix can be calculated as usual by

$$[K]^{(e)} = \int_{v^{(e)}} [B]^T [D][B] dv \qquad (2.35)$$

Since quantities are independent of θ, integration on θ from 0 to 2π can be performed. Thus

$$[K]^{(e)} = \iint_{\Omega^{(e)}} [B]^T [D][B] 2\pi r dr dz \qquad (2.36)$$

As the element becomes smaller and smaller, the variation of r and z becomes very small compared to the value of r, and hence $[B]$ can be considered to be constant within the element. As first approximation stiffness matrix can be determined by considering the

integrand to be constant equal to its value at the centroid of the triangle and multiply it with area of triangle.

Hence stiffness matrix is

$$[K]^{(e)} = [\overline{B}]^T [D][\overline{B}] 2\pi \overline{r} A \tag{2.37}$$

where $[\overline{B}]^T$ is the matrix $[B]$ evaluated at the centroid and A is the area of triangle having nodes i,j,k.

In axisymmetric problems forces are also distributed axisymmetrically. Load vector due to body force $\{\overline{R} \quad \overline{Z}\}^T$ can be calculated using natural coordinates. For triangular ring type element elemental load vector for body forces is given by:

$$\{P_b^{(e)}\} = \frac{\pi A}{12} \begin{Bmatrix} (2r_i + r_j + r_k)\overline{R} \\ (2r_i + r_j + r_k)\overline{Z} \\ (r_i + 2r_j + r_k)\overline{R} \\ (r_i + 2r_j + r_k)\overline{Z} \\ (r_i + r_j + 2r_k)\overline{R} \\ (r_i + r_j + 2r_k)\overline{Z} \end{Bmatrix} \tag{2.38}$$

Load vector due to traction $\{T_r \quad T_z\}^T$ and assuming ij-side of the element coincides with traction boundary can be given as:

$$\{P\}_S^{(e)} = \frac{\pi S_{ij}}{3} \begin{Bmatrix} (2r_i + r_j)T_r \\ (2r_i + r_j)T_z \\ (r_i + 2r_j)T_r \\ (r_i + 2r_j)T_z \\ 0 \\ 0 \end{Bmatrix} \tag{2.39}$$

where S_{ij} is the length of side ij.

Using FEM, thermal problems can also be solved following similar steps as that of solving solid mechanics problems. In thermal problems having convective boundary condition element stiffness matrix gets added with another matrix known as element

stiffness matrix due to convective boundary condition $[K]_h^{(e)}$. For axisymmetric triangular ring type element having ij side on the convective boundary, stiffness matrix due to convective boundary condition is given by:

$$[K]_h^{(e)} = 2\pi r_{sij} h \begin{bmatrix} \frac{1}{3} & \frac{1}{6} & 0 \\ \frac{1}{6} & \frac{1}{3} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{2.40}$$

where h is convective heat transfer coefficient and r is the radial distance of convective boundary.

Thermal load vector for convective heat transfer problem is given by:

$$\{P\}_S^{(e)} = \frac{\pi S_{ij}}{3} \begin{Bmatrix} (2r_i + r_j)h\phi_\infty \\ 0 \\ (r_i + 2r_j)h\phi_\infty \\ 0 \\ 0 \\ 0 \end{Bmatrix} \tag{2.41}$$

here S_{ij} is the length of side ij which is on the convective boundary, ϕ_∞ is the temperature of the fluid and h is the convective heat transfer coefficient between fluid and surface.

Next section reviews the application of above theory to calculate thermal stresses in axisymmetric solids.

2.6 FEM Analysis for Thermal Stresses in Axisymmetric Solids

Some times an elastic solid is subjected to deformation due to effects like temperature change or moisture absorption etc. For example, the strain in a 1-D rod due to a temperature change ΔT can be given as:

$$\varepsilon_0 = \alpha \Delta T \tag{2.42}$$

where α is the thermal expansion coefficient, and ΔT is the temperature change. For triangular axisymmetric element thermal strain is given by:

$$\{\varepsilon_0\} = \{\alpha\Delta T \quad \alpha\Delta T \quad \alpha\Delta T \quad 0\}^T \tag{2.43}$$

If the body is free to expand initial strain will develop due to temperature change and if it is constraint to expand initial stresses will develop. A combined expression for state of stress considering both initial stress and initial strain can be given as:

$$\{\sigma\} = [D](\{\varepsilon\} - \{\varepsilon_0\}) + \{\sigma_0\} \tag{2.44}$$

where $\{\sigma_0\}$ and $\{\varepsilon_0\}$ are initial stress and strain respectively.

Element stiffness matrix $\left[K^{(e)}\right]$ can be calculated as usual by:

$$[K]^{(e)} = \int_{v^{(e)}} [B]^T [D][B] dv \tag{2.45}$$

Following the potential energy formulation given in previous section one can get thermal load vector as:

$$\left\{P_{th}^{(e)}\right\} = \int_{v^{(e)}} [B]^T \{\sigma_0\} dv \tag{2.46}$$

For very fine mesh thermal load vector can be given by:

$$\left\{P_{th}^{(e)}\right\} = 2\pi \begin{bmatrix} b_i & \frac{N_i}{\bar{r}} & 0 & c_i \\ 0 & 0 & c_i & b_i \\ b_j & \frac{N_j}{\bar{r}} & 0 & c_j \\ 0 & 0 & c_j & b_j \\ b_k & \frac{N_k}{\bar{r}} & 0 & c_k \\ 0 & 0 & c_k & b_k \end{bmatrix} \begin{bmatrix} 1-\nu & \nu & \nu & 0 \\ \nu & 1-\nu & \nu & 0 \\ \nu & \nu & 1-\nu & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \frac{E}{(1+\nu)(1-2\nu)} \begin{Bmatrix} \alpha \\ \alpha \\ \alpha \\ 0 \end{Bmatrix} \Delta T \bar{r} A \tag{2.47}$$

at centroid $N_i = N_j = N_k = \frac{1}{3}$.

In the present project temperature and stress response of the pipe material through which steam is flowing at constant pressure with constant flow rate, is simulated for linear change in the temperature of the steam. Next section reviews the transient thermal analysis that is necessary to solve the problem [1,2].

2.7 Transient Thermal Analysis

The difference between structural and nonstructural dynamic problems is in the order of governing differential equation. Compared to structural analysis these problems deal only with the first derivative in time. Governing differential equation for heat transfer problem is given as [1]:

$$[K]\{\phi\}+[C]\{\dot{\phi}\}=\{F\} \tag{2.48}$$

Here ϕ is unknown temperature, F is force vector and for very fine mesh $[C]$ can be given as:

$$[C]=\frac{\pi c\rho A\bar{r}}{6}\begin{bmatrix}2 & 1 & 1\\ 1 & 2 & 1\\ 1 & 1 & 2\end{bmatrix} \tag{2.49}$$

where c and ρ are specific heat and density of the material respectively. A is the area of the element and $\bar{r}$ is the radial distance of the centroid of the element.

There are many numerical schemes to solve above equations. In the present project Neumark method is used which is unconditionally stable. Neumark method can be summarized in following steps:

1. Form the $[K]^{(e)}$ and $[C]^{(e)}$ matrices.
2. Calculate ϕ and $\dot{\phi}$ from the initial condition i.e. from initial steady state condition prevailing before transient.
3. Select time step Δt and parameter α and δ and calculate following integration constants:

$$a_0=\frac{1}{\alpha\Delta t^2} \qquad a_1=\frac{\delta}{\alpha\Delta t} \qquad a_2=\frac{1}{\alpha\Delta t}$$

$$a_3 = \frac{1}{2\alpha} - 1 \qquad a_4 = \frac{\delta}{\alpha} - 1 \qquad a_5 = \frac{\Delta t}{2}\left(\frac{\delta}{\alpha} - 2\right)$$

$$a_6 = \Delta t(1-\delta) \qquad a_7 = \delta \Delta t$$

4. Form effective stiffness matrix $[\hat{K}] = [K] + a_1[C]$.

1. Calculate the effective load at time $t + \Delta t$

$$\hat{R}^{t+\Delta t} = R^{t+\Delta t} + [C](a_1 \phi^t + a_4 \dot{\phi}^t)$$

2. Solve for temperature at time $t + \Delta t$:

$$\{\phi^{t+\Delta t}\} = [\hat{K}]^{-1} \hat{R}^{t+\Delta t}$$

In the present project $\hat{R}^{t+\Delta t}$ at *t=0* is the force vector in the steady state i.e. before transient starts.[1]

2.8 Fatigue Monitoring

It has been recognized that a metal subjected to a repetitive or fluctuating stress will fail at a stress much lower than that required causing failure on a single application of load. Failures occurring under conditions of dynamic loading are called fatigue failures, because these failures occur only after a considerable period of service. This effect is the most dominant when the component approaches the end of the designed life. To avoid such failures on-line monitoring of critical power plant component is done. In the present work Miner Rule is used to calculate the fatigue usage of the component, which relates the cumulative damage under cyclic stressing to net work absorbed by the component. According to this rule the number of stress cycles applied, expressed as a percent of the number to failure at the given stress level, would be the proportion of useful life expended. When the total damage defined by this concept reaches 100%, the component should fail.

At any time, say t, if a component has seen $n_1, n_2, n_3 \ldots\ldots n_n$ number of stress cycles with maximum stress $\sigma_1, \sigma_2, \sigma_3, \ldots\ldots \sigma_n$ respectively and if corresponding to these stress

levels the fatigue life of the component is $N_1, N_2, N_3, \ldots\ldots N_n$, then fatigue usage of the component at time t is

$$\frac{n1}{N1} + \frac{n2}{N2} + \frac{n3}{N3} + \ldots\ldots\ldots \frac{nn}{Nn} \qquad (2.49)$$

Due to the statistical nature of the fatigue it is recommended that when fatigue usage of the component reaches 0.9, component should be replaced [3].

In the present work it is assumed that electronic sensors acquiesce steam temperature round the clock and prepares a data file *temp.txt* that stores all the deviations in the steam temperature and duration of the deviation. At 12 O'clock in the night every day one code get executed which reads all the entries in *temp.txt* and calculates the fatigue usage of the components as shown in section 4.4.Same code writes fatigue usage of the component, date wise in file *usage.txt*.

Chapter 3

Artificial Neural Network

3.1 Introduction

The field of Artificial Neural Networks (ANN) was born in the 1940's when McCulloch and Pitts proposed a computational model based on a simple neuron – like logical element. Donald Hebb described a learning rule for adapting the connection strengths of these artificial neurons [12]. The Hebb rule or delta rule became the basis of almost all ANN research that was to follow. After a period of relative inactivity from early 50's to late 70's a resurgence of interest came in early 80's, with the work of Hopfield and his associates as well as Rumelhart, Hinton and Williams who developed a generalization of the delta rule which made possible the training of powerful multi–layered learning networks [12].

In the recent years ANNs have been applied to many problems, including pattern recognition and classification, associative memory, functional optimization, phoneme recognition, word spotting, talker identification, language recognition, signal separation, time series prediction and combinatorial optimization. The decade has also shown much progress in the development of new learning algorithms, structured neural networks and unsupervised learning. The next section reviews the Error Back-propagation Algorithm, which is the most commonly used algorithm.

3.2 Error Back-propagation Algorithm

Network used for back propagation algorithm is shown in Fig. 3.1. The network functions as follows: Each neuron receives a signal from the neurons in the previous layer, and each of those signals is multiplied by a separate weight value. The weighted inputs are summed, and passed through a limiting function, which scales the output to a fixed range

of values. The output of the limiter is then broadcast to all of the neurons in the next layer. So, to use the network to solve a problem, the input values are applied to the input neurons of the first layer, the signals are allowed to propagate through the network, and the output values are calculated.

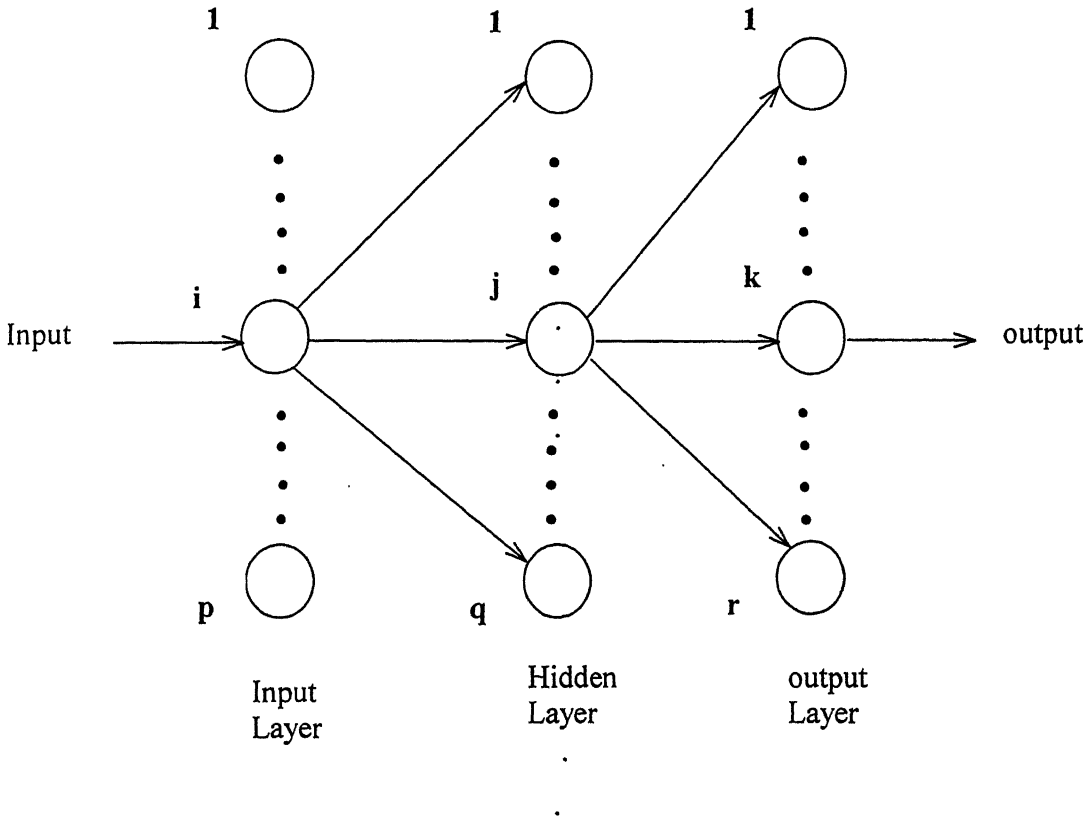


Fig. 3.1 Network Topology for Back-propagation algorithm

Since the real uniqueness or 'intelligence' of the network exists in the values of the weights between neurons, we need a method of adjusting the weights to solve a particular problem. For this type of network, the most common learning algorithm is called Back Propagation (BP). A BP network learns by example, that is, we must provide a learning set that consists of some input examples and the known-correct output for each case. So, we use these input-output examples to show the network what type of behavior is expected, and the BP algorithm allows the network to adapt.

In BP algorithm essentially, the idea is to perform a gradient descend on the error of the network:

$$E = \frac{1}{2} \sum_{j=1}^{N} (t_j - y_j)^2 \tag{3.1}$$

where N is the total no. of output nodes, t_j is the desired target value and y_j denotes net's response to an input vector or pattern. The weights are adjusted to reduce the difference of net's response and the desired target value according to the delta rule by the following formula:

$$\Delta w_{j,i} = \eta f'(a_j)(t_j - y_j) x_i \tag{3.2}$$

where $f'(a_j)$ represents how quickly activation changes due to excitation of input x_i and η represents learning rate. In general, the delta rule may be written as:

$$\Delta w_{j,i} = \eta \delta_j x_i \tag{3.3}$$

for output nodes:

$$\delta_j = f'(a_j)(t_j - y_j) \tag{3.4}$$

The errors of output nodes than further propagate backwards to the units in the hidden layer(s) and contribute to the change in weights. Consider a hidden node k and let I_k be the set of nodes which node k connects to (I_k is the fan-out of k), therefore taking into account the contribution to the error from all of these nodes in I_k. Hence error signal for hidden unit becomes:

$$\delta_k = f'(a_k) \sum_{j \in I_k} \delta_j w_{j,k} \tag{3.5}$$

Above weight updation continues till the error reaches below desired value .[5,12].

Simple feed forward networks using BP algorithm can be trained to accomplish pattern recognition tasks with complex non-linear boundaries, they are limited to processing static patterns – patterns those are fixed rather than temporal in nature. To overcome this limitation time delay neural networks and adaptive time delay neural networks are proposed [4,8,11].

3.3 Training and Testing Data Generation

3.3.1 Training Data Generation

Training data is generated using FEM code, which converts transient in the steam temperature to stress response of the pipe material. FEM code is executed for following linear changes in the steam temperature taking over time span of 2 to 10 minutes from normal operating temperature i.e. 400 ^{0}C: 360 ^{0}C, 380 ^{0}C, 420 ^{0}C, 440 ^{0}C, 460 ^{0}C. When operator comes to know about deviation in the steam temperature he takes some corrective actions and steam temperature returns to its normal operating value i.e. 400 ^{0}C. It is assumed that the temperature of the steam returns linearly and takes same time as taken in the deviation from the normal operating temperature. Fig. 3.3.1.1 shows the steam temperature profile during transient. In this figure ΔT is the change in the steam temperature from normal operating temperature before operator takes corrective measures to bring the temperature to the normal operating value. In training data generation ΔT = 20^{0}C, 40^{0}C, 60 ^{0}C, -20 ^{0}C, -40 ^{0}C. t is transient buildup time for the steam temperature and its values are taken as 2 to 8 minutes for training data generation. For each case the value of maximum and minimum principal stresses, and their times (in seconds) are noted down at inner and outer radii of the pipe. These values are given in for table 3.3.1.1.

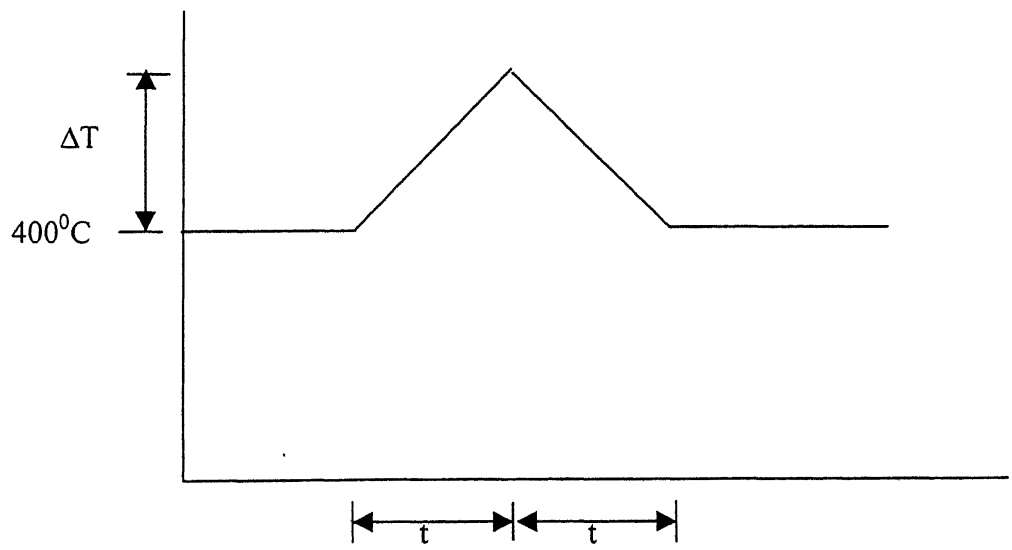


Fig. 3.3.1.1 Steam temperature profile during transient

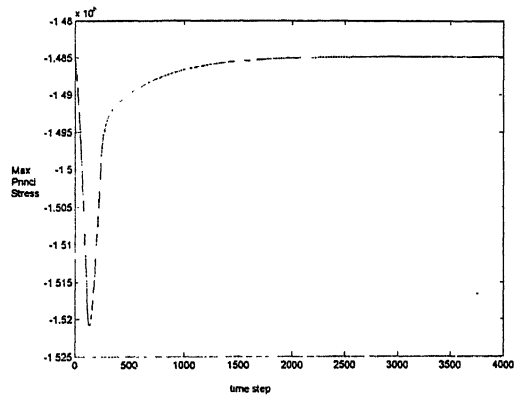
Table 3.3.1.1 Training Data Generated by FEM

Temp Change (degC)	Tran Time (min)	Inner radius				Outer radius			
		Max Stress		Min stress		Max Stress		Min stress	
		Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)
+20	2	-1.4849	1	-1.5208	136	-1.4522	73	-1.4685	286
	3	-1.4849	1	-1.5247	199	-1.4525	73	-1.4746	383
	4	-1.4849	1	-1.5276	262	-1.4527	73	-1.4795	482
	5	-1.4849	1	-1.5301	326	-1.4528	73	-1.4836	568
	6	-1.4849	1	-1.5328	390	-1.4528	73	-1.4871	647
	7	-1.4849	1	-1.5341	454	-1.4529	73	-1.4902	726
	8	-1.4849	1	-1.5359	517	-1.4529	73	-1.4929	799
	9	-1.4849	1	-1.5374	581	-1.4529	73	-1.4953	871
	10	-1.4849	1	-1.5389	644	-1.4529	73	-1.4975	942
+40	2	-1.4849	1	-1.5569	136	-1.4514	73	-1.4840	286
	3	-1.4849	1	-1.5645	199	-1.4520	73	-1.4961	383
	4	-1.4849	1	-1.5704	262	-1.4522	73	-1.5060	482
	5	-1.4849	1	-1.5753	326	-1.4524	73	-1.5141	568
	6	-1.4849	1	-1.5796	390	-1.4525	73	-1.5211	647
	7	-1.4849	1	-1.5835	454	-1.4525	73	-1.5273	726
	8	-1.4849	1	-1.5869	517	-1.4527	73	-1.5327	799
	9	-1.4849	1	-1.5900	581	-1.4527	73	-1.5375	871
	10	-1.4849	1	-1.5929	644	-1.4528	73	-1.5418	942
+60	2	-1.4849	1	-1.5926	136	-1.4505	73	-1.4994	286
	3	-1.4849	1	-1.6043	199	-1.4514	73	-1.5177	383
	4	-1.4849	1	-1.6132	262	-1.4518	73	-1.5324	482
	5	-1.4849	1	-1.6206	326	-1.4521	73	-1.5446	568
	6	-1.4849	1	-1.6270	390	-1.4522	73	-1.5551	647
	7	-1.4849	1	-1.6328	454	-1.4524	73	-1.5644	726
	8	-1.4849	1	-1.6379	517	-1.4524	73	-1.5725	799
	9	-1.4849	1	-1.6426	581	-1.4525	73	-1.5797	871
	10	-1.4849	1	-1.6469	644	-1.4526	73	-1.5826	942
-20	2	-1.4489	136	-1.4849	1	-1.4377	286	-1.4540	73
	3	-1.4450	199	-1.4849	1	-1.4316	383	-1.4537	73
	4	-1.4421	262	-1.4849	1	-1.4267	482	-1.4535	73
	5	-1.4396	326	-1.4849	1	-1.4226	568	-1.4535	73
	6	-1.4375	390	-1.4849	1	-1.4134	647	-1.4534	73
	7	-1.4352	454	-1.4849	1	-1.4145	726	-1.4533	73
	8	-1.4338	517	-1.4849	1	-1.4133	799	-1.4533	73
	9	-1.4323	581	-1.4849	1	-1.4109	871	-1.4533	73
	10	-1.4308	644	-1.4849	1	-1.4088	942	-1.4533	73
-40	2	-1.4130	136	-1.4849	1	-1.4223	286	-1.4548	73
	3	-1.4052	199	-1.4849	1	-1.4101	383	-1.4543	73
	4	-1.3993	262	-1.4849	1	-1.4002	482	-1.4540	73
	5	-1.3944	326	-1.4849	1	-1.3921	568	-1.4538	73

	6	-1.3901	390	-1.4849	1	-1.3857	647	-1.4537	73
	7	-1.3860	454	-1.4849	1	-1.3789	726	-1.4536	73
	8	-1.328	517	-1.4849	1	-1.3735	799	-1.4535	73
	9	-1.3797	581	-1.4849	1	-1.3687	871	-1.4535	73
	10	-1.3768	644	-1.4849	1	-1.3644	942	-1.4535	73

3.3.2 Sample Plots for Training

Following are some of the sample plots obtained by FEM showing the transient maximum principal thermal stress at inner and outer radius of the pipe:



Fig,.3.3.2.1 Max. Principal Thermal Stress at inner radius for+20^{0}C change in steam temperature (for 2 min transient)

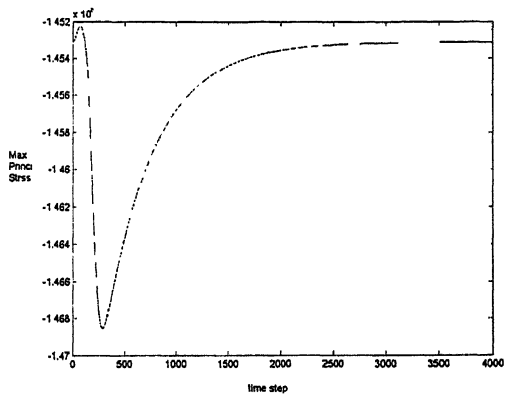


Fig 3.3.2.2 Max. Principal Thermal Stress at outer radius for +20^{0}C change in steam temperature (for 2 min transient).

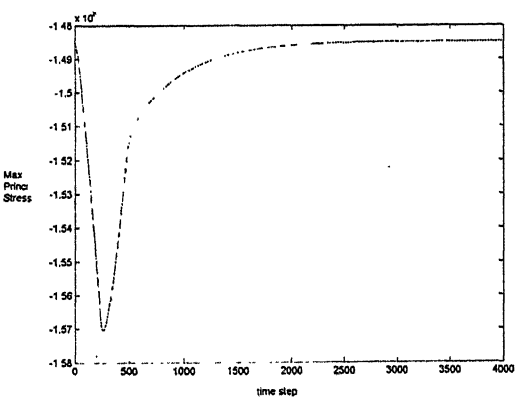


Fig 3.3.2.3 Max. Principal Thermal Stress at inner radius for+40^{0}C change in steam temperature (for 4 min transient)

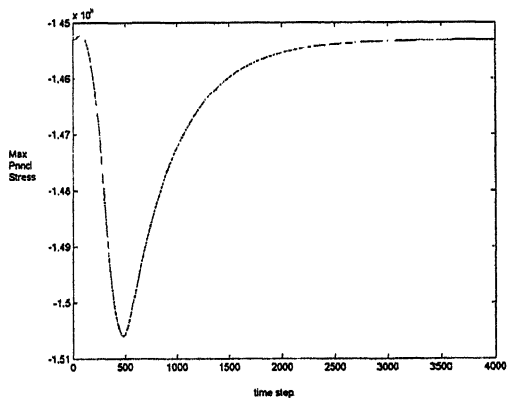


Fig 3.3.2.4 Max. Principal Thermal Stress at outer radius for +40^{0}C change in steam temperature (for 4 min transient).

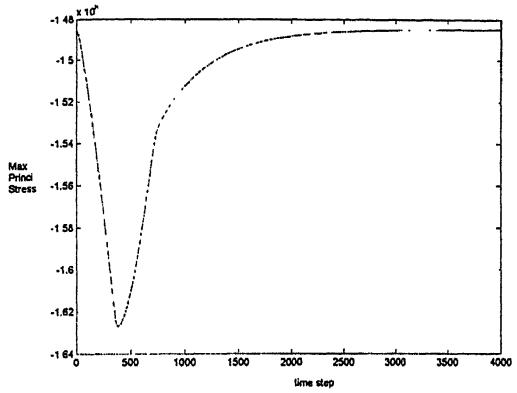


Fig 3.3.2.5 Max. Principal Thermal Stress at inner radius for+60^{0}C change in steam temperature (for 6 min transient)

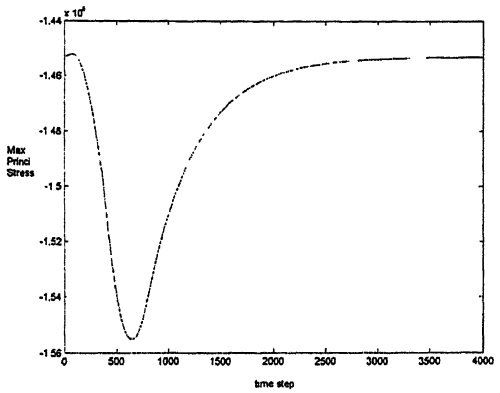


Fig 3.3.2.6 Max. Principal Thermal Stress at outer radius for +60^{0}C change in steam temperature (for 6 min transient).

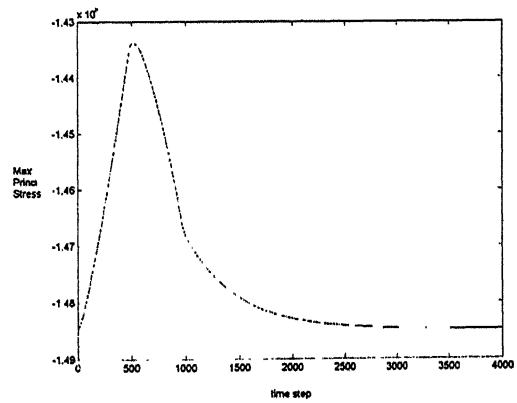


Fig 3.3.2.7 Max. Principal Thermal Stress at inner radius for -20^{0}C change in steam temperature (for 8 min transient)

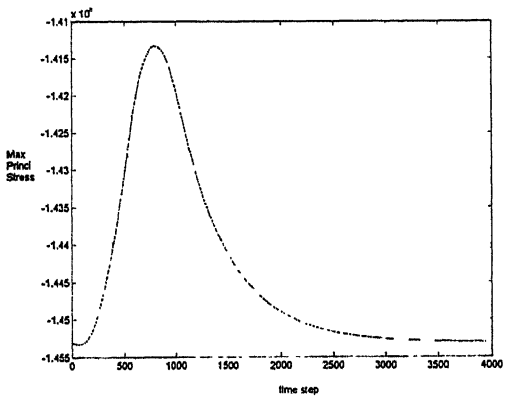


Fig 3.3.2.8 Max. Principal Thermal Stress at outer radius for -20^{0}C change in steam temperature (for 8 min transient).

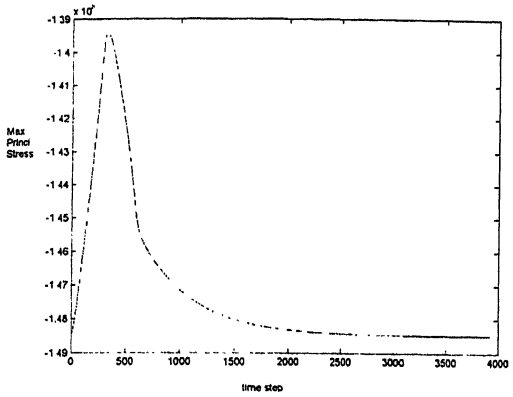


Fig 3.3.2.9 Max. Principal Thermal Stress at inner radius for -40^{0}C change in steam temperature (for 5 min transient)

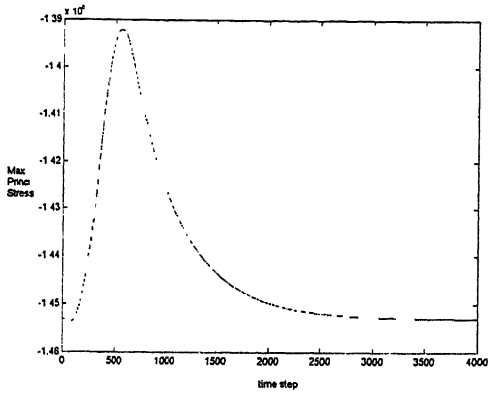


Fig 3.3.2.10 Max. Principal Thermal Stress at outer radius for -40^{0}C change in steam temperature (for 5 min transient).

3.3.3 Testing Data Generation

The same FEM Code that generates training data for the input parameters during training also generates testing data. Testing data is generated for $\Delta T = 30^0$C, 50^0C, 70 ^{0}C, -30 ^{0}C, -50 ^{0}C and t = 3min, 4min, 5min and 6min. These parameters lie out of the parameters used for generating training data. Testing data is given in table below in the same way as the training data and will be used to compare the simulation results of neural network in the next chapter.

Table 3.3.3.1 Testing Data Generated by FEM

Temp Change (degC)	Tran Time (min)	Inner radius				Outer radius			
		Max Stress		Min stress		Max Stress		Min stress	
		Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)
+30	3	-1.4849	1	-1.5446	199	-1.4522	73	-1.4854	383
	4	-1.4849	1	-1.5490	262	-1.4525	73	-1.4928	482
	5	-1.4849	1	-1.5527	326	-1.4526	73	-1.4989	567
	6	-1.4849	1	-1.5590	390	-1.4527	73	-1.5041	647
+50	3	-1.4849	1	-1.5244	199	-1.4517	73	-1.5069	383
	4	-1.4849	1	-1.5918	262	-1.4520	73	-1.5192	482
	5	-1.4849	1	-1.5980	326	-1.4522	73	-1.5293	567
	6	-1.4849	1	-1.6033	390	-1.4524	73	-1.5381	647

+70	3	-1.4849	1	-1.6242	199	-1.4511	73	-1.5284	383
	4	-1.4849	1	-1.6346	262	-1.4516	73	-1.5456	482
	5	-1.4849	1	-1.6432	326	-1.4516	73	-1.5598	567
	6	-1.4849	1	-1.6507	390	-1.4521	73	-1.5722	647
-30	3	-1.4251	199	-1.4849	1	-1.4208	383	-1.4540	73
	4	-1.4207	262	-1.4849	1	-1.4135	482	-1.4538	73
	5	-1.4170	326	-1.4849	1	-1.4074	567	-1.4536	73
	6	-1.4138	390	-1.4849	1	-1.4021	647	-1.4535	73
-50	3	-1.3853	199	-1.4849	1	-1.3993	383	-1.4545	73
	4	-1.3779	262	-1.4849	1	-1.3870	482	-1.4542	73
	5	-1.3717	326	-1.4849	1	-1.3769	567	-1.4540	73
	6	-1.3664	390	-1.4849	1	-1.3681	647	-1.4538	73

Chapter 4

Results and Discussion

4.1 Parameters

ANN with back propagation learning paradigm is used to train the network to match input-output relationships using the training cases discussed in the previous chapter. Following parameter values are used:

4.1.1 Initial Weights

The weights of the network to be trained are typically initialized at small random values. The initialization strongly affects the ultimate solution. If all the weights start out with equal values, and if the solution requires that the unequal weights be developed, the network may not train properly. Unless the network is disturbed by random factors or random character of input patterns during training. The choice of initial weights is, however one of the several factors affecting the training of the network towards an acceptable error minimum.

4.1.2 Learning Constant

The effectiveness and convergence of the error back-propagation learning algorithm depends significantly on the values of the weight learning rate constants i.e. η_1. In general, however, the optimum value of learning rates depends on the problem being solved and there is no single learning constant value suitable for different training cases.

4.1.3 Momentum Factor

The purpose of momentum factor is to accelerate the convergence of the training. The factor involves supplementing the current weight adjustment with a fraction of the most recent weight adjustments.

4.2 Training Exit Criterion

Training of the neural network stops if any one of the following conditions is met during training:

- Error level reaches a pre-specified error value.
- All specified training epochs get exhausted.

4.3 Results

A feed forward back propagation neural network with the following network topology was trained with the training data generated by FEM Module and given in the table 3.3.1.1.

A three-layered fully connected NN (one input layer, one hidden layer and one output layer) was configured with two input neurons taking temperature change and duration of the transient as input to the NN, eighteen hidden neurons and eight output neurons giving maximum and minimum stress and time (in sec) of their occurrence at both inner and outer radii. All the neurons use sigmoidal squashing function. Other network parameters are given below:

Learning Rate for weights = 0.4

Momentum Factor for weight updation = 0.9

Maximum no. of epochs = 50,000

Network is trained in batch mode.

Fig 4.3.1 to Fig 4.3.3 show variation of mean squared error (MSE) vs. no. of epochs starting from different random initial weights, as training proceeds. Its obvious from the error plots that even if the starting error is different for all the three cases but in 50,000 epochs training network converges to approximately same error level.

Table 4.3.1 shows the simulation results for the testing data shown in obtained in previous chapter. Negative errors show under-prediction.

In the present work the data is normalized linearly between maximum and minimum values of parameters. While predicting time instants for maximum and minimum stresses, some of the time values came out to be negative indicating that stresses become maximum or minimum before the start of the transient. To avoid such situations following mapping rules are applied in addition to normalization of the data:

- In the prediction of time of occurrence for maximum or minimum stresses at inner radius, if predicted time is less than 100 seconds it is mapped at 1s.
- In the prediction of time of occurrence for maximum or minimum stresses at outer radius, if predicted time is less than 200 seconds it is mapped at 73s.

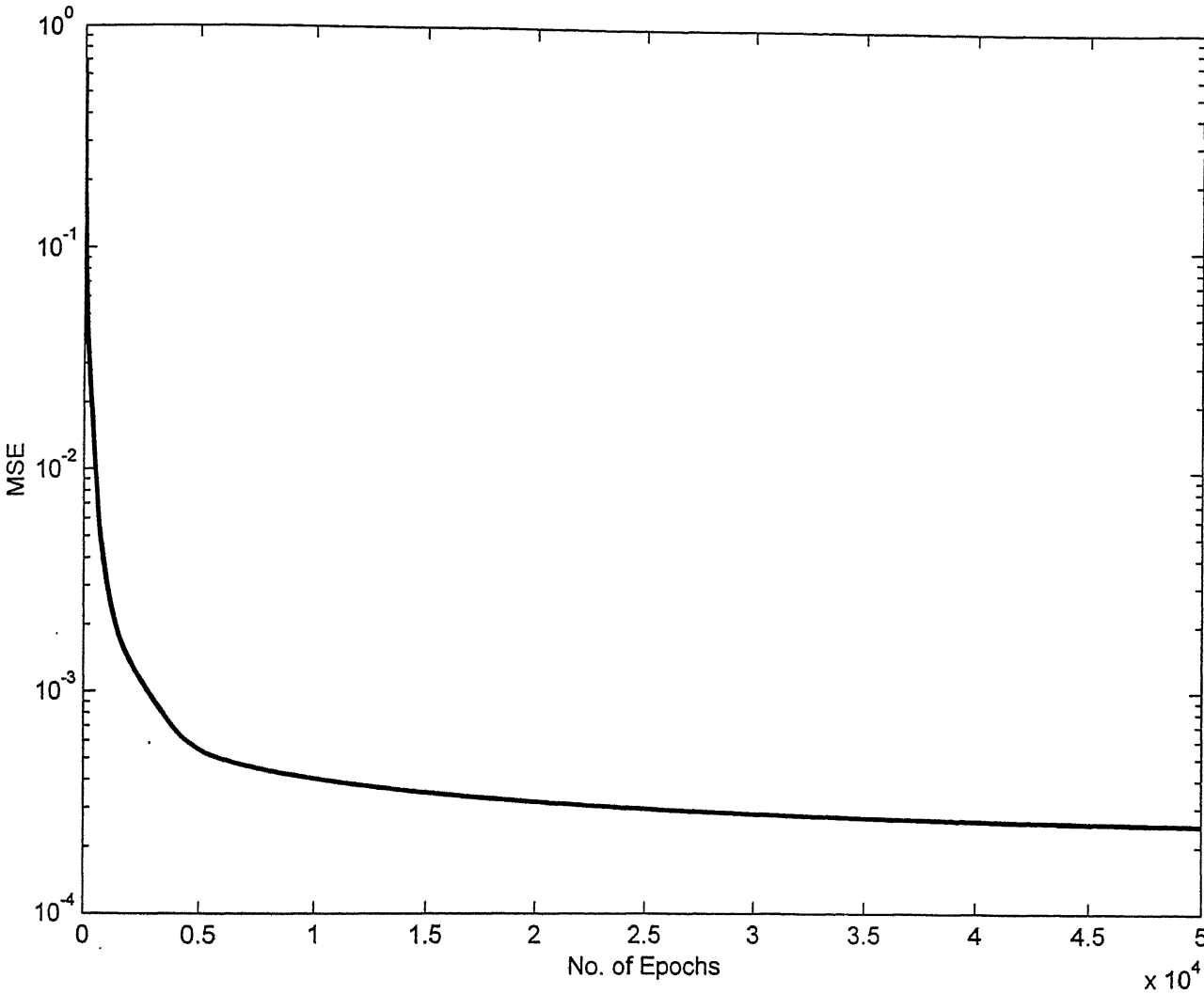


Fig 4.3.1 MSE vs. No. of Epochs for first set of initial weights

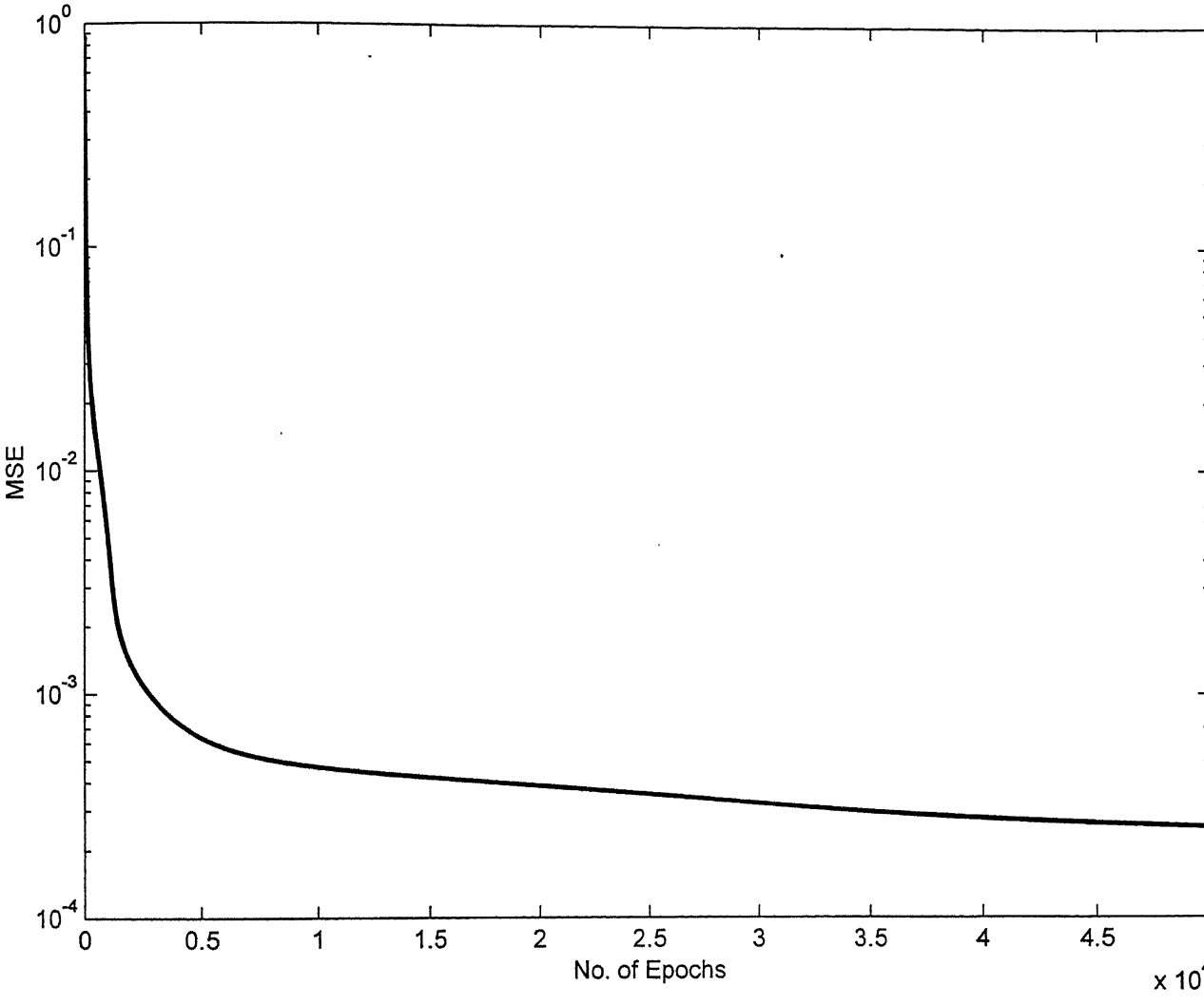


Fig 4.3.2 MSE vs. No. of Epochs for second set of initial weights

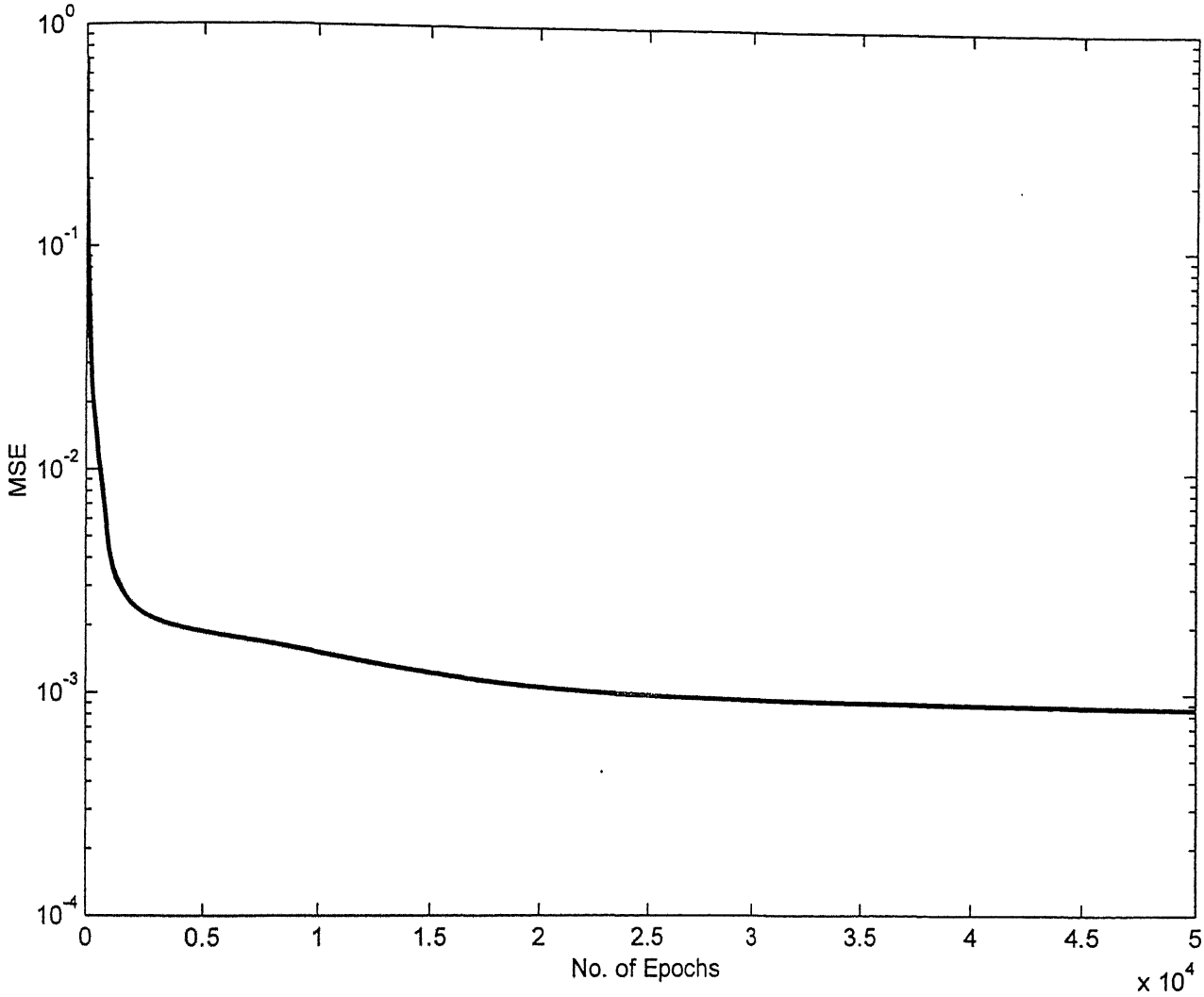


Fig 4.3.3 MSE vs. No. of Epochs for third set of initial weights

Table 4.3.1 Simulation results for Testing Data

Temp Change (degC)	Tran Time (min)		Inner radius				Outer radius			
			Max Stress		Min stress		Max Stress		Min stress	
			Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)
+30	3	Predicted	-1.4832	1	-1.5360	205	-1.4535	73	-1.4808	394
		Exact	-1.4849	1	-1.5446	199	-1.4522	73	-1.4854	382
		%Error	-0.11	0	-0.5	3.05	0.089	0	-0.31	3.14
	4	Predicted	-1.4832	1	-1.5519	270	-1.4531	73	-1.4958	483
		Exact	-1.4849	1	-1.5490	262	-1.4525	73	-1.4928	481
		%Error	-0.11	0	0.18	3.05	0.04	0	-0.13	0.41
	5	Predicted	-1.4816	1	-1.5640	306	-1.4504	73	-1.5065	536
		Exact	-1.4849	1	-1.5527	326	-1.4526	73	-1.4989	566
		%Error	-0.22	0	0.72	-6.13	-0.15	0	-0.33	-5.3
	6	Predicted	-1.4803	1	-1.5688	363	-1.4477	73	-1.5132	615
		Exact	-1.4849	1	-1.5590	391	-1.4527	73	-1.5041	646
		%Error	-0.30	0	0.62	-7.16	0.34	0	0.60	4.78
+50	3	Predicted	-1.4851	1	-1.5937	182	-1.4498	73	-1.5095	365
		Exact	-1.4849	1	-1.5844	199	-1.4517	73	-1.5069	383
		%Error	0.01	0	0.58	8.54	-0.13	0	-0.11	-4.7
	4	Predicted	-1.4809	1	-1.5995	261	-1.4497	73	-1.5269	487
		Exact	-1.4849	1	-1.5918	262	-1.4520	73	-1.5192	482
		%Error	-0.02	0	-0.77	-0.38	-0.15	0	0.506	1.04
	5	Predicted	-1.4893	1	-1.5871	333	-1.4549	73	-1.5238	575
		Exact	-1.4849	1	-1.5980	326	-1.4522	73	-1.5293	567
		%Error	-0.29	0	1.09	2.14	0.18	0	-0.35	1.41
	6	Predicted	-1.4874	1	-1.5929	393	-1.4538	73	-1.5325	648
		Exact	-1.4849	1	-1.6033	391	-1.4524	73	-1.5381	647
		%Error	0.16	0	-0.64	0.51	0.096	0	-0.36	0.15
+70	3	Predicted	-1.4954	1	-1.5980	183	-1.4545	73	-1.5044	353
		Exact	-1.4849	1	-1.6242	199	-1.4511	73	-1.5284	383
		%Error	0.70	0	-1.61	8.04	0.23	0	-1.57	-7.8
	4	Predicted	-1.4937	1	-1.6135	285	-1.4571	73	-1.5248	494
		Exact	-1.4849	1	-1.6346	262	-1.4516	73	-1.5456	482
		%Error	0.59	0	1.29	8.77	0.37	0	-1.34	2.5
	5	Predicted	-1.4777	1	-1.6443	368	-1.4513	73	-1.5610	617
		Exact	-1.4849	1	-1.6432	326	-1.4516	73	-1.5598	567
		%Error	-0.48	0	0.066	12.88	-0.02	0	0.07	8.81
	6	Predicted	-1.4679	1	-1.6744	431	-1.4442	73	-1.5902	705
		Exact	-1.4849	1	-1.6507	391	-1.4521	73	-1.5722	647
		%Error	-0.11	0	1.41	10.23	-0.54	0	1.14	8.96
-30	3	Predicted	-1.4212	211	-1.4853	1	-1.4170	406	-1.4553	73
		Exact	-1.4251	199	-1.4849	1	-1.4208	383	-1.4540	73
		%Error	0.26	6.03	0.026	0	-0.26	6.00	0.089	0
	4	Predicted	-1.4232	265	-1.4847	1	-1.4140	485	-1.4539	73
		Exact	-1.4207	262	-1.4849	1	-1.4135	482	-1.4538	73
		%Error	0.16	1.15	-0.013	0	0.042	0.622	0.006	0
	5	Predicted	-1.4312	355	-1.4814	1	-1.4153	603	-1.4500	73

		Exact	-1.4170	326	-1.4849	1	-1.4074	567	-1.4536	73
		%Error	1.00	8.89	-0.23	0	0.56	6.34	-0.24	0
	6	Predicted	-1.4245	441	-1.4812	1	-1.4090	708	-1.4497	73
		Exact	-1.4138	391	-1.4849	1	-1.4021	647	-1.4535	73
		%Error	0.75	12.78	-0.25	0	0.49	9.4	-0.26	0
-50	3	Predicted	-1.3810	163	-1.4855	1	-1.3940	319	-1.4560	73
		Exact	-1.3853	199	-1.4849	1	-1.3993	383	-1.4545	73
		%Error	-0.31	-18.1	0.04	0	-0.37	-16.1	0.10	0
	4	Predicted	-1.3678	279	-1.4849	1	-1.3785	483	-1.4557	73
		Exact	-1.3779	262	-1.4849	1	-1.3870	482	-1.4542	73
		%Error	-0.73	6.5	0	0	-0.61	0.20	0.10	0
	5	Predicted	-1.3545	304	-1.4905	1	-1.3679	521	-1.4608	73
		Exact	-1.3717	326	-1.4849	1	-1.3769	567	-1.4540	73
		%Error	-1.15	6.74	0.37	0	-0.65	-8.11	0.46	0
	6	Predicted	-1.3508	342	-1.4928	1	-1.3645	585	-1.4619	73
		Exact	-1.3664	390	-1.4849	1	-1.3681	647	-1.4538	73
		%Error	0.83	-14.0	0.53	0	-0.26	-9.5	0.55	0

4.4 Pattern Memorization by Neural Network

In neural networks it seems that lower the error level reached in training, the testing results will be better. But some times if the error in training reaches below a certain level, network looses generalization capability and is said to be over trained. In that case simulation results for the pattern used for the training are very accurate but those for the patterns not seen during the training are very poor. This is what happened in the present problem when network was trained with Levenberg-Marquardt Back-propagation Algorithm. This algorithm uses adaptive learning rate and converses very fast and in 2500 epochs only the Mean Square Error (MSE) reached to the order of 10^{-7}, but with this error level network got over trained. Simulation results for the training patterns were very close to the exact values obtained by the FEM module but those for the pattern not seen in the training were very poor. Another network with same parameters other than the exit error criteria (training gets stopped when Mean Square Error (MSE) reaches 0.001) was trained with Levenberg-Marquardt Back-propagation Algorithm. Network reached the desired error level in less than 150 epoch training. Fig 4.4.1 and fig 4.4.2 gives plots for Mean Square Error (MSE) as training progresses for over training and adequate training cases respectively. The simulation results for the training patterns and patterns not seen during training with this network are with in the acceptable error range.

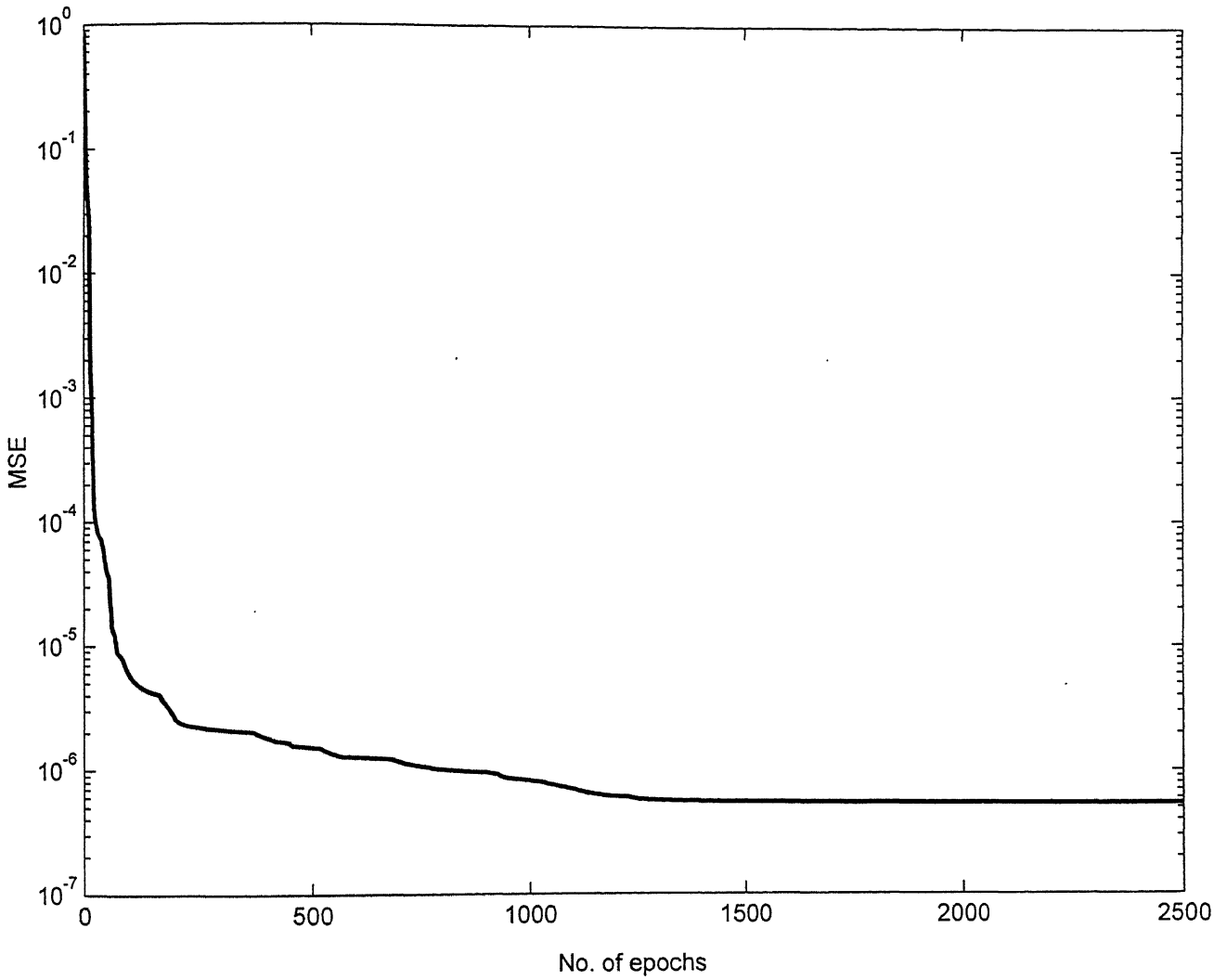


Fig 4.4.1 MSE vs. No. of Epochs for over training of the network

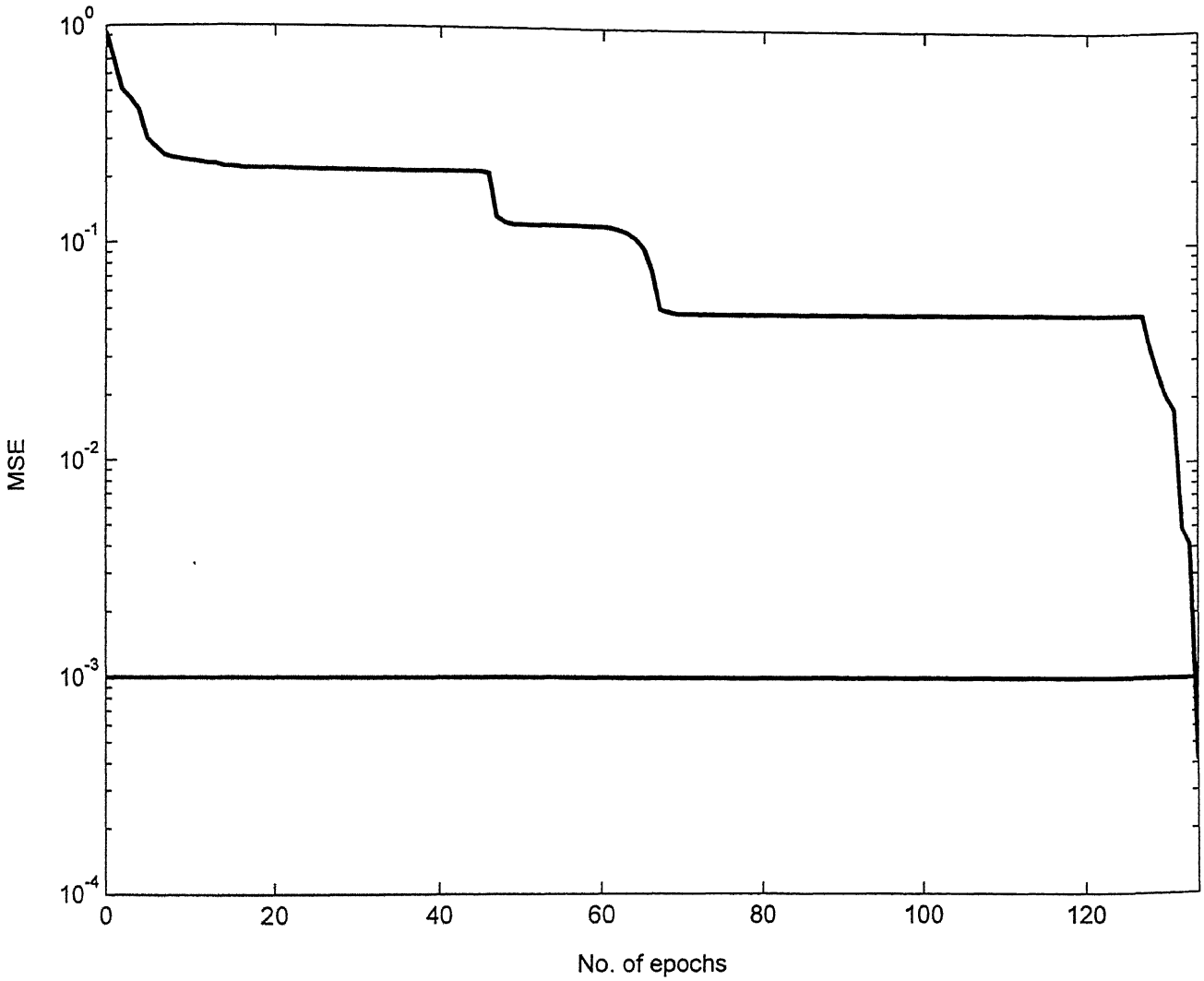


Fig 4.4.2 MSE vs. No. of Epochs for adequate training of the network

4.4.1 Simulation Results for Over Training Case with Training Patterns as Input

Fig 4.4.1.1 to Fig.4.4.1.4 show some of the simulation results for the patterns seen during training. Single hashed bar shows the exact value obtained by FEM while double hashed bar shows value predicted by ANN respectively. It can be observed from the figures that the network follows the original values nicely.

Maximum error level in the prediction of various parameters is given in the following table 4.4.1.1. Only the magnitude of the error is given not sign.

Table 4.4.1.1 Max. error in the prediction of various parameters when training patterns are used as input parameters for over training case

Parameter	Max. Percentage Error in simulation
Max. Principal Stress at inner radius.	0.0207
Time of occurrence of Max. Principal Stress at inner radius.	0.7353
Min. Principal Stress at inner radius.	0.0377
Time of occurrence of Min. Principal Stress at inner radius.	0.7353
Max. Principal Stress at outer radius.	0.0141
Time of occurrence of Max. Principal Stress at outer radius.	0.6224
Min. Principal Stress at outer radius.	0.0696
Time of occurrence of Min. Principal Stress at outer radius.	0.6833

Principal stresses shown in the following figures are compressive in nature. Here only the magnitude of the max. principal stress is given not sign.

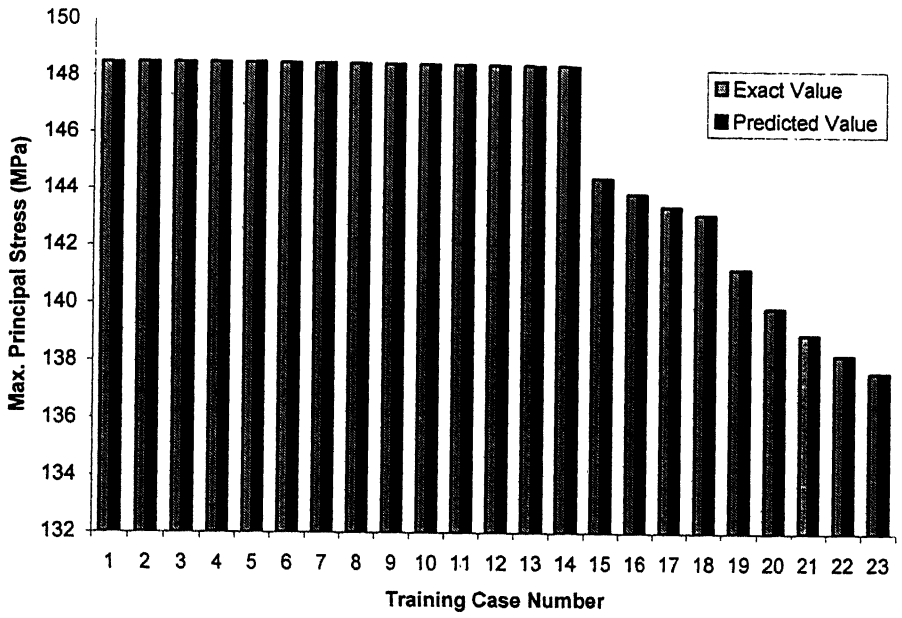


Fig 4.4.1.1 Simulation results for maximum of Maximum Principal Stress at inner radius

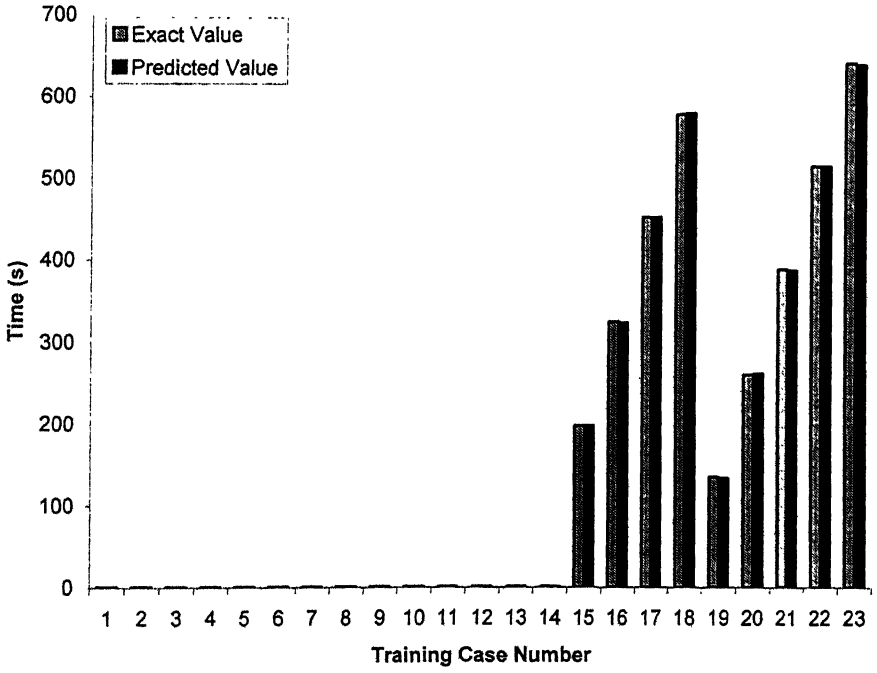


Fig 4.4.1.2 Simulation results for time of occurrence of maximum of Maximum Principal Stress at inner radius.

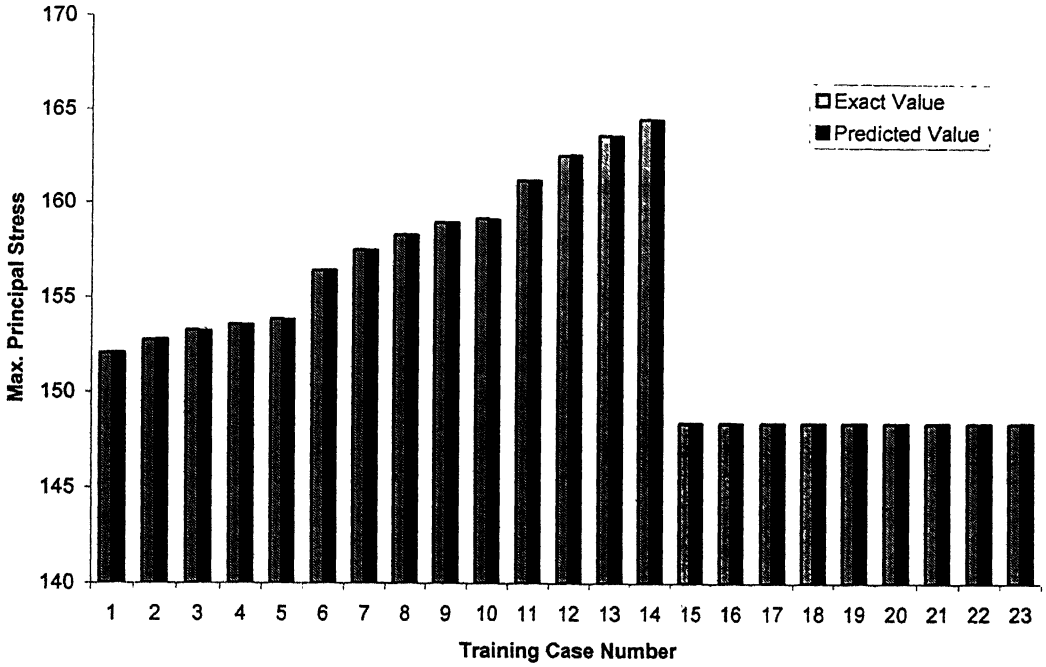


Fig 4.4.1.3 Simulation results for minimum of Maximum Principal Stress at inner radius

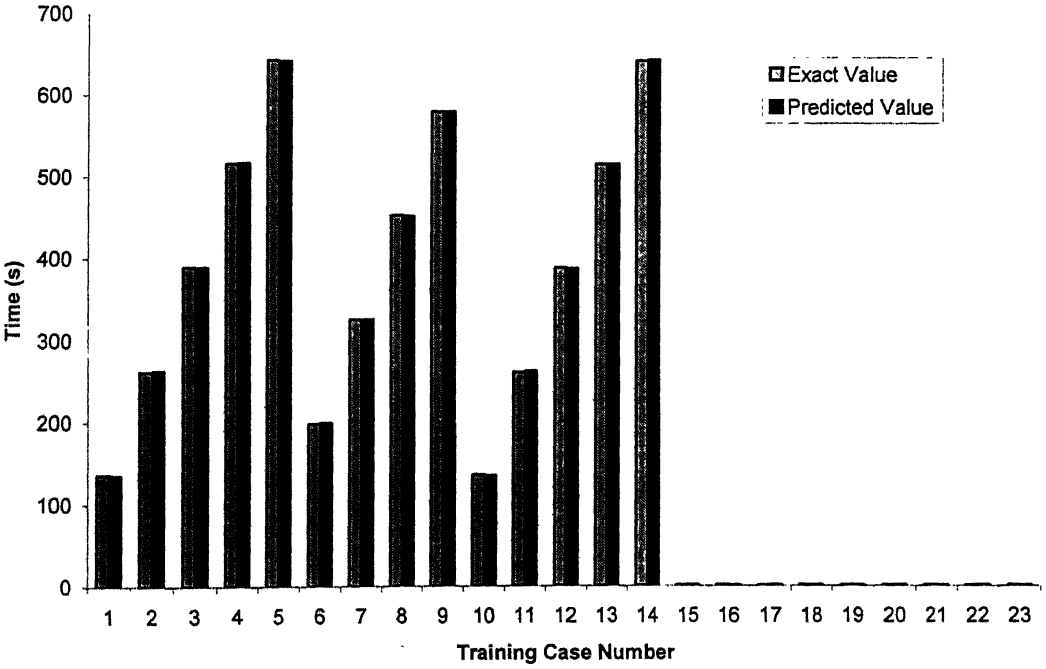


Fig 4.4.1.4 Simulation results for time of occurrence of minimum of Maximum Principal Stress at inner radius

4.4.2 Simulation Results for Over Training Case with Test Patterns as Input

Fig 4.4.2.1 to Fig.4.4.1.4 show some of the simulation results for the test patterns i.e patterns not seen during training. Single hashed bar shows the exact value obtained by FEM while double hashed bar shows value predicted by ANN respectively. It can be observed from the figures that the network's performance is very poor in this case.

Maximum error level in the prediction of various parameters is given in the following table 4.4.1.1. Only the magnitude of the error is given not sign.

Table 4.4.2.1 Max. error in the prediction of various parameters when testing patterns are used as input parameters for over training case.

Parameter	Max. Percentage Error in simulation
Max. Principal Stress at inner radius.	20.9
Time of occurrence of Max. Principal Stress at inner radius.	9.9e3
Min. Principal Stress at inner radius.	17.41
Time of occurrence of Min. Principal Stress at inner radius.	7.62e3
Max. Principal Stress at outer radius.	16.05
Time of occurrence of Max. Principal Stress at outer radius.	1.92e3
Min. Principal Stress at outer radius.	24.37
Time of occurrence of Min. Principal Stress at outer radius.	1.99e3

Principal stresses shown in the following figures are compressive in nature. Here only the magnitude of the max. principal stress is given not sign.

Principal stresses shown in the following figures are compressive in nature. Here only the magnitude of the max. principal stress is given not sign.

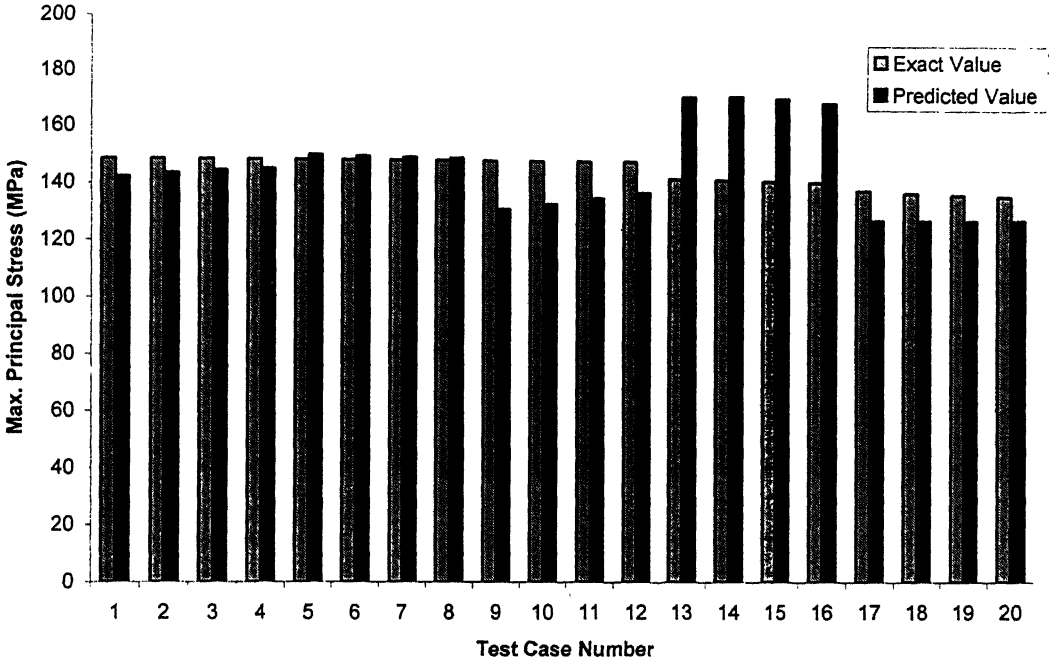


Fig 4.4.2.1 Simulation results for maximum of Maximum Principal Stress at inner radius

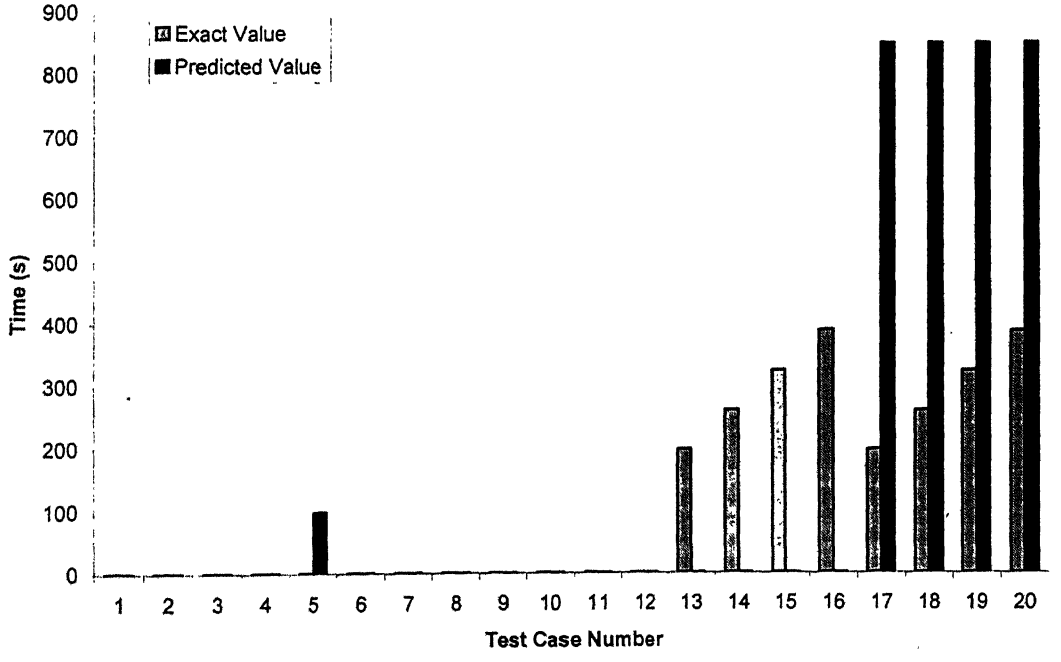


Fig 4.4.2.2 Simulation results for time of occurrence of maximum of Maximum Principal Stress at inner radius.

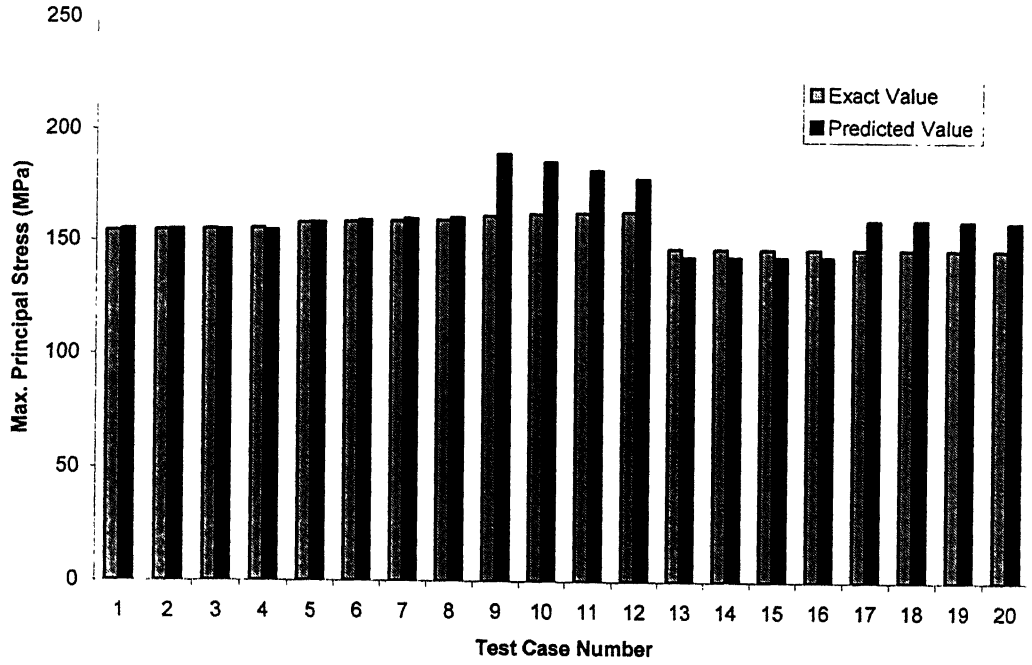


Fig 4.4.2.3 Simulation results for minimum of Maximum Principal Stress at inner radius

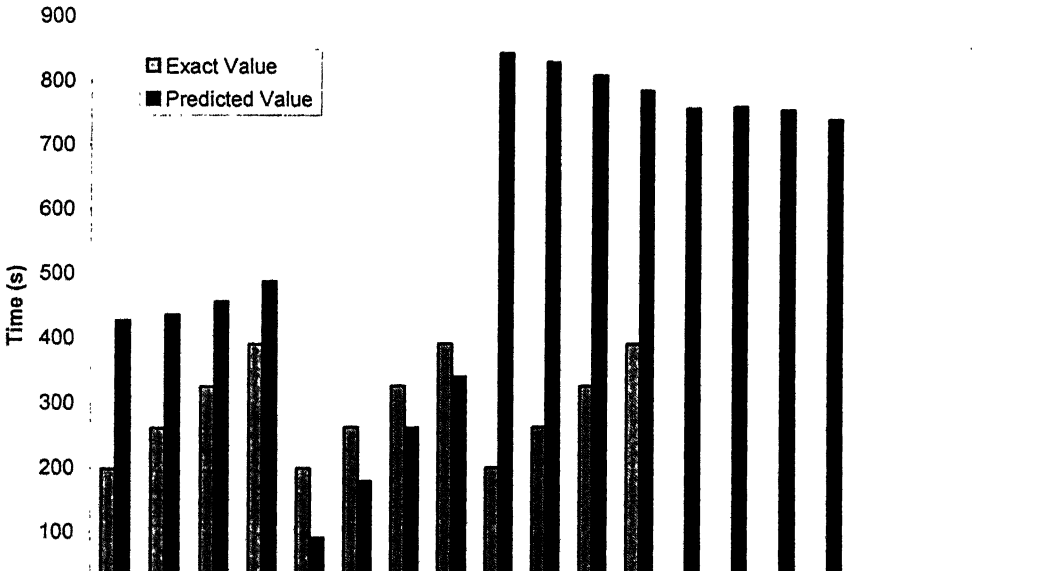


Fig 4.4.2.4 Simulation results for time of occurrence of minimum of Maximum Principal Stress at inner radius

Its clear from the previous sections that when MSE level is of the order of 10^{-7}, simulation results for training patterns are very accurate and percentage error is always

less than 1%, but those for test data are very poor. Hence, at this MSE level network lost its generalization capability. Next section gives the simulation results for adequate training case i.e. when MSE is of the order of 10^{-2}.

4.3.3 Simulation Results for adequate Training Case with Test Patterns as Input

Fig 4.4.3.1 to Fig.4.4.3.8 show the simulation results for the test patterns i.e patterns not seen during training for adequate training case. Single hashed bar shows the exact value obtained by FEM while double hashed bar shows value predicted by ANN respectively. It can be observed from the figures that the network's performance is very poor in this case.

Maximum error level in the prediction of various parameters is given in the following table 4.4.3.1. Only the magnitude of the error is given not sign.

Table 4.4.3.1 Max. error in the prediction of various parameters when test patterns are used as input parameters for over training case.

Parameter	Max. Percentage Error in simulation
Max. Principal Stress at inner radius.	1.08
Time of occurrence of Max. Principal Stress at inner radius.	14.57
Min. Principal Stress at inner radius.	1.42
Time of occurrence of Min. Principal Stress at inner radius.	13.06
Max. Principal Stress at outer radius.	1.18
Time of occurrence of Max. Principal Stress at outer radius.	11.48
Min. Principal Stress at outer radius.	3.04
Time of occurrence of Min. Principal Stress at outer radius.	11

Principal stresses shown in the following figures are compressive in nature. Here only the magnitude of the max. principal stress is given not sign.

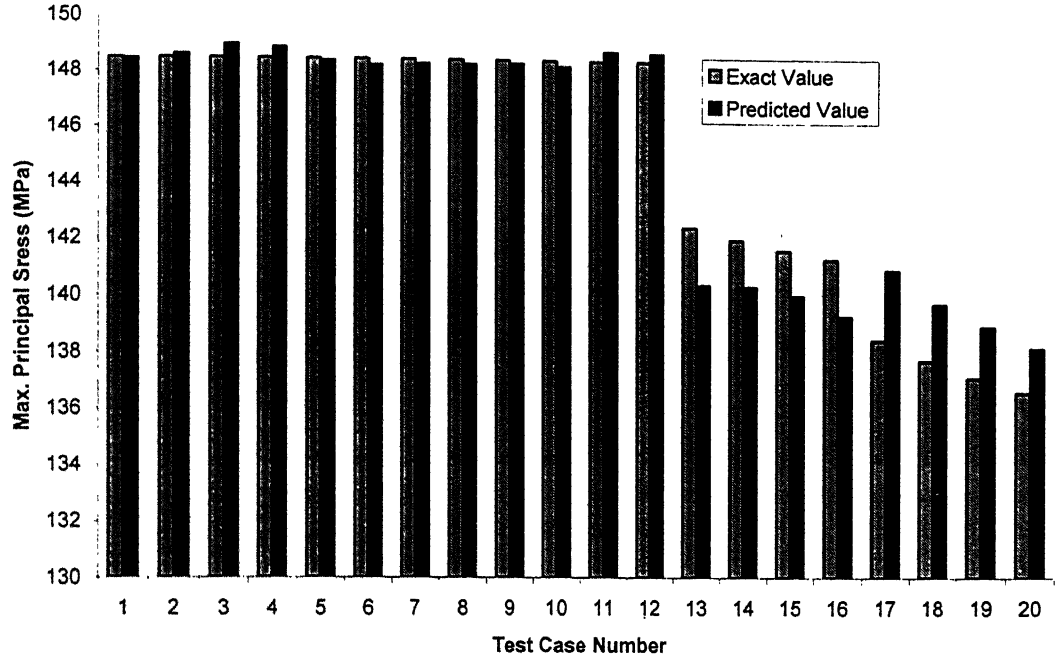


Fig 4.4.3.1 Simulation results for maximum of Maximum Principal Stress at inner radius

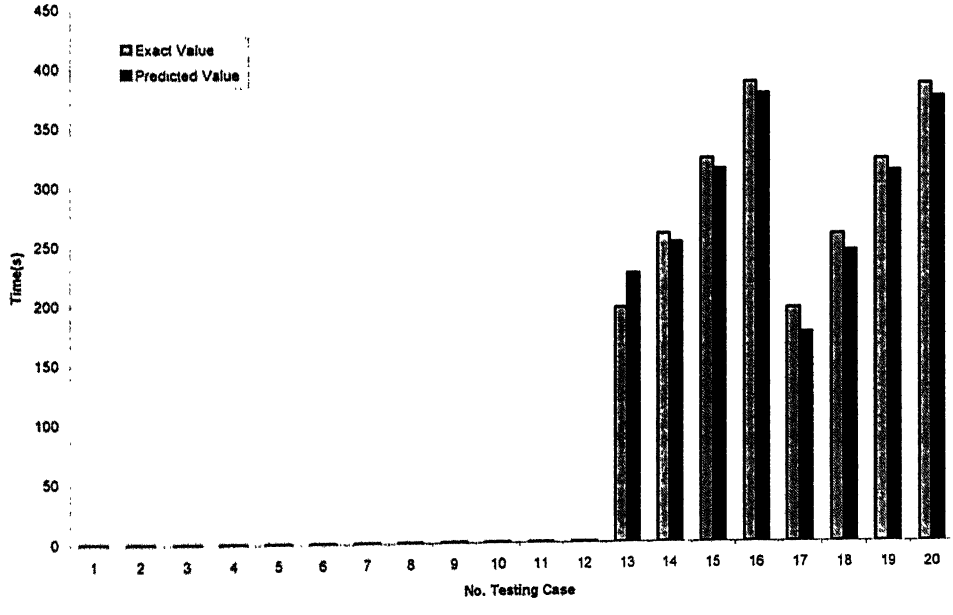


Fig 4.4.3.2 Simulation results for time of occurrence of maximum of Maximum Principal Stress at inner radius.

Principal stresses shown in the following figures are compressive in nature. Here only the magnitude of the max. principal stress is given not sign.

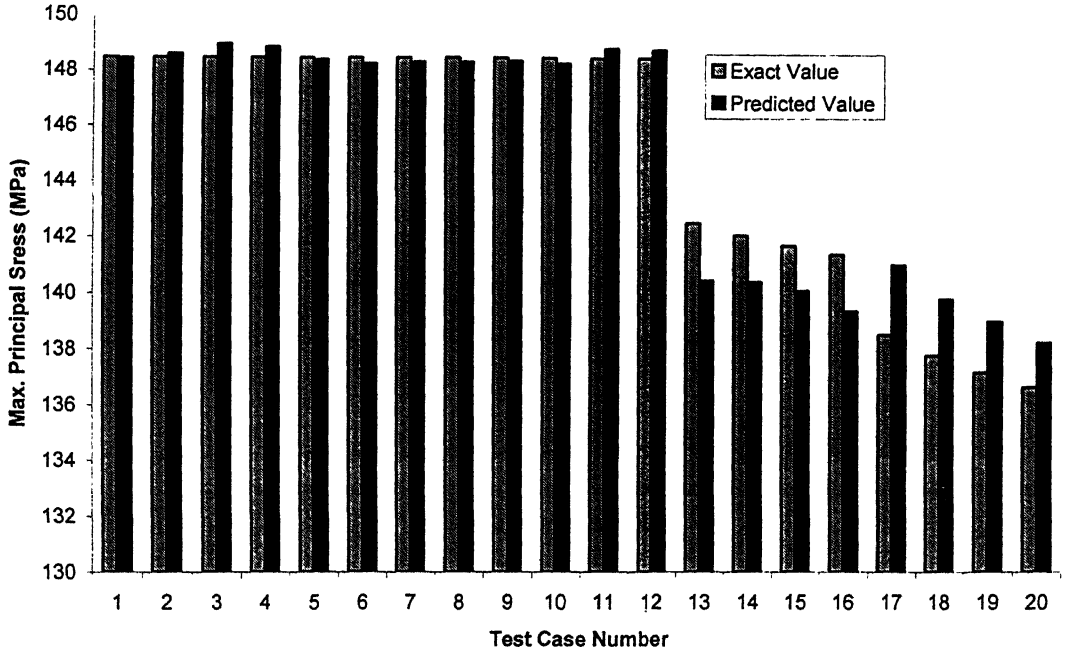


Fig 4.4.3.1 Simulation results for maximum of Maximum Principal Stress at inner radius

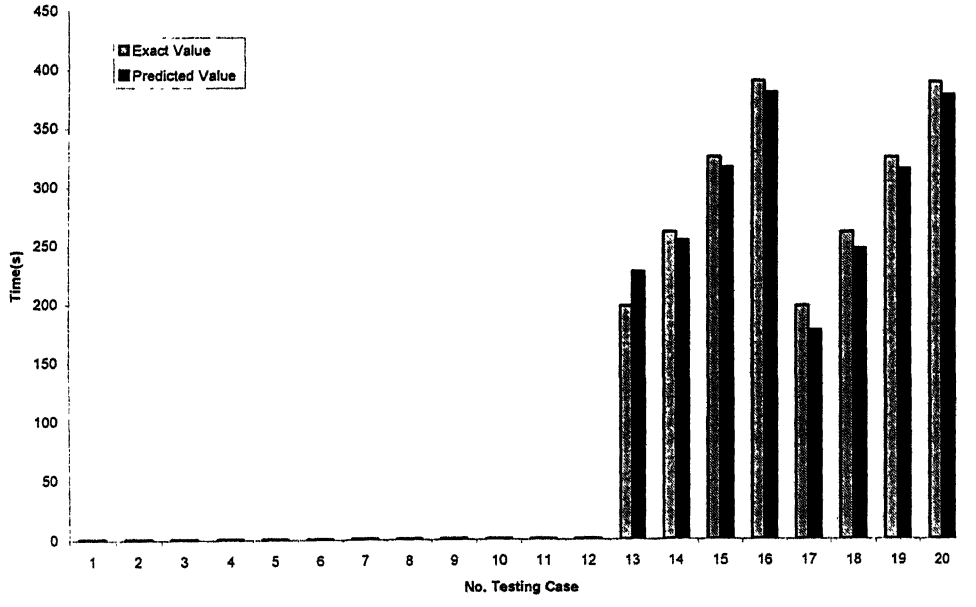


Fig 4.4.3.2 Simulation results for time of occurrence of maximum of Maximum Principal Stress at inner radius.

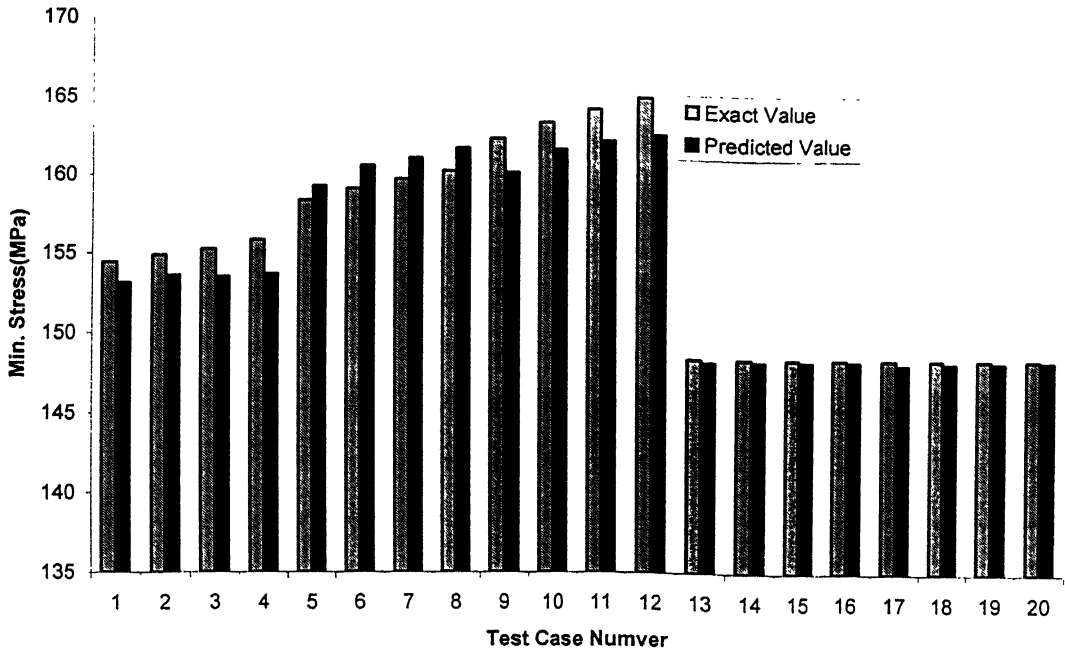


Fig 4.4.3.3 Simulation results for minimum of Maximum Principal Stress at inner radius

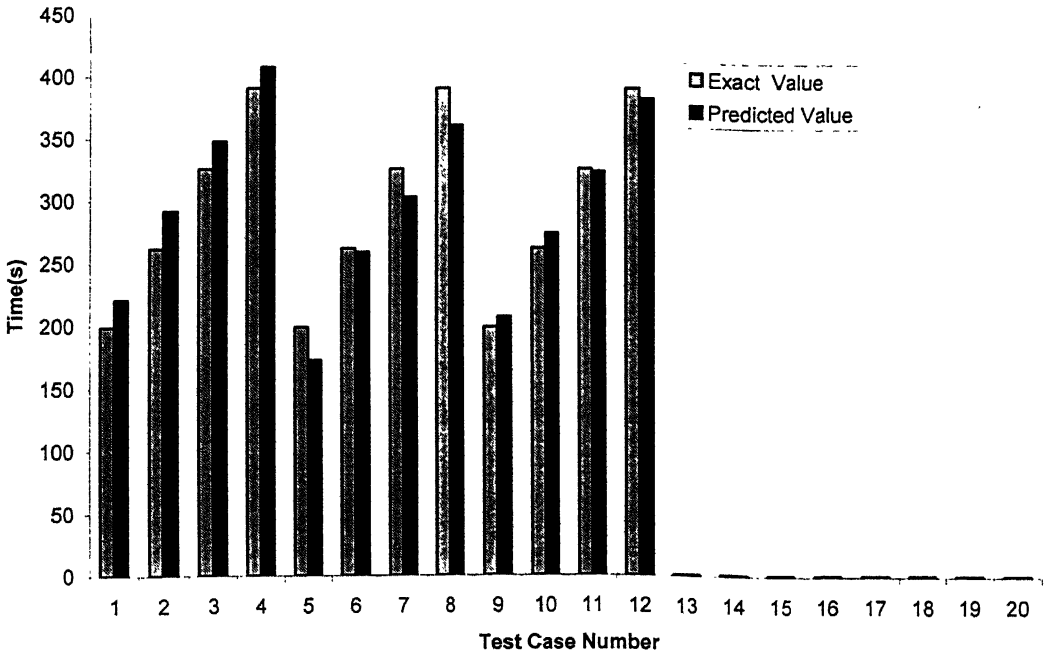


Fig 4.4.3.4 Simulation results for time of occurrence of minimum of Maximum Principal Stress at inner radius.

From the results of the section 4.4.1, section 4.4.2 and section 4.4.3 its obvious that over training the network deteriorates its capability to generalize input-output relationship.

4.4 Fatigue Usage of the Component

In all the above-simulated cases stress level is below the endurance limit of the material, hence these plant transients will have no effect on the fatigue life of the component and component can survive for infinite such stress cycles. But for the demonstration purpose let us assume that fatigue life corresponding to 20^0C, 40^0C and 60^0C is $N_1 = 10^6$ cycles, $N_2 = 10^5$ cycles and $N_3 = 10^4$ cycles respectively. At any particular day if the entries in the data acquisition file are:

Temp Change (ΔT^0 C)	20	40	60
No. of Cycles (n_i)	4,500	4,370	2,860

It shows that by that day component has seen 104500 transients in the steam temperature of intensity 20^0C i.e. steam temperature fluctuated from 400 to 420^0C, 4370 transients in the steam temperature of intensity 40^0C and 2860 transients in the steam temperature of intensity 60^0C

Using equation (2.49) and the above N_i and n_i values, code calculates the fatigue usage of the component to be

$$\sum_{i=1}^{3} \frac{n_i}{N_i} = 0.3342$$

Assuming that maximum usage factor of the component before replacement is 0.9, at the end of the day remaining fatigue usage is only 0.5608.

Chapter 5

Conclusion and Future Scope

This work was carried out to study the feasibility of using Artificial Neural Network in FEM based On-Line Fatigue Monitoring system developed by BARC. Following conclusions are drawn from the analysis of various results obtained in previous chapter:

- Error Back-Propagation Neural Network can be used to predict maximum principal stress and its time of occurrence in thick pipe, through which steam is flowing at high temperature, for transient change in steam temperature. Maximum percentage error in the prediction of various parameters as:

	Max.% error	Corresponding to Change in Steam Temperature (ΔT^0C)
Max. Princi. Stress at inner radius	1.15	-50
Time of occurrence of Max Princi..Stress at inner radius	18.09	-50
Min. Princi. Stress at inner radius	1.62	70
Time of occurrence of Min Princi..Stress at inner radius	12.88	70
Max. Princi. Stress at outer radius	0.65	-50
Time of occurrence of Max Princi..Stress at outer radius	16.71	-50
Min. Princi. Stress at outer radius	1.5703	70
Time of occurrence of Min Princi..Stress at outer radius	8.9645	70

- Maximum percentage error magnitude is either corresponding to ΔT (Change in steam from normal operating temperature before operator takes a corrective measure) equal to 70^0C or -50^0C, which lie out of range of ΔT used for training the network. This shows that network is poor in extrapolation. Hence, full possible range of input parameters should be used to train the network.

- Error is more in prediction of time. Maximum error found is about 18%, However in the present problem, we are more concerned with the stress level values then the time when the maximum stress occur. In predicting maximum stress values, the error is less than 2%, which may be well acceptable, considering other sources of errors which may arise from measurements etc. In predicting fatigue life another important aspect is the number of times maximum stress exceeds the fatigue strength according to S-N diagram. Since the developed ANN predicts the maximum stress close to that obtained by FEM, reliable prediction of fatigue life may be possible with developed ANN.

- For the present problem if error level during training is reduced beyond a certain level, network gets over trained i.e. it starts memorizing training patterns and looses generalization capability for input / output relationship. In present problem value of MSE of the order of 10^{-3} is adequate for training LM Back-propagation Neural Network.

5.1 Scope for the Future Work

Problem solved is essentially temporal in nature. In the present work error back-propagation algorithm is used, which gives excellent results in processing static patterns. In future temporal pattern processing networks like time –delay neural network or adaptive time-delay neural network can be used to solve the problem.

Appendix-A

Simulation Result Tables

Table 1, Table 2 and Table 3 give the simulation results for over training case with training patterns as input, with test patterns and those of with the test patterns respectively. Negative errors show under-prediction of the data.

Table A.1 Simulation results for training patterns (over training case)

Temp Change (degC)	Tran Time (min)		Inner radius				Outer radius			
			Max Stress		Min stress		Max Stress		Min stress	
			Value (MPa)	Time (s)	Value (MPa)	Time (s)	Value (MPa)	Time (s)	Value (MPa)	Time (s)
+20	2	Predicted	-148.47	1	-152.09	135	-145.23	73	-146.89	288
		Exact	-148.49	1	-152.08	136	-145.22	73	-146.85	286
		% Error	-0.0135	0	0.0066	-0.74	0.0069	0	0.0272	0.699
	3	Predicted	-148.48	1	-152.48	199	-145.25	73	-147.45	386
		Exact	-148.49	1	-152.47	199	-145.25	73	-147.46	383
		% Error	-0.0067	0	0.0066	0	0	0	-0.0068	0.783
	4	Predicted	-148.49	1	-152.78	263	-145.27	73	-147.93	479
		Exact	-148.49	1	-152.76	262	-145.27	73	-147.95	482
		% Error	0	0	0.0131	0.38	0	0	-0.0135	-0.622
	5	Predicted	-148.49	1	-153.03	326	-145.28	73	-148.35	565
		Exact	-148.49	1	-153.01	326	-145.28	73	-148.36	568
		% Error	0	0	0.0131	0	0	0	-0.0067	-0.528
	6	Predicted	-148.48	1	-153.23	390	-145.28	73	-148.71	647
		Exact	-148.49	1	-153.28	390	-145.28	73	-148.71	647
		% Error	-0.0067	0	-0.0326	0	0	0	0	0
	7	Predicted	-148.49	1	-153.42	453	-145.29	73	-149.05	723
		Exact	-148.49	1	-153.41	454	-145.29	73	-149.02	726
		% Error	0	0	0.0065	-0.22	0	0	0.0201	-0.413
	8	Predicted	-148.49	1	-153.58	518	-145.28	73	-149.31	799
		Exact	-148.49	1	-153.59	517	-145.29	73	-149.29	799
		% Error	0	0	-0.0065	0.19	-0.0069	0	0.0134	0
	9	Predicted	-148.49	1	-153.73	582	-145.29	73	-149.54	872
		Exact	-148.49	1	-153.74	581	-145.29	73	-149.53	871
		% Error	0	0	-0.0065	0.17	0	0	0.0067	0.115
	10	Predicted	-148.50	1	-153.88	643	-145.31	73	-149.73	942
		Exact	-148.49	1	-153.89	644	-145.29	73	-149.75	942

		% Error	0.0067	0	-0.0065	-0.15	0.0138	0	-0.0134	0
+40	2	Predicted	-148.48	1	-155.68	135	-145.14	73	-148.45	285
		Exact	-148.49	1	-155.69	136	-145.14	73	-148.40	286
		% Error	-0.0067	0	-0.0064	-0.74	0	0	0.0337	-0.35
	3	Predicted	-148.49	1	-156.46	200	-145.20	73	-149.60	386
		Exact	-148.49	1	-156.45	199	-145.20	73	-149.61	383
		% Error	0	0	0.0064	0.5	0	0	-0.0067	0.783
	4	Predicted	-148.50	1	-157.04	263	-145.23	73	-150.56	480
		Exact	-148.49	1	-157.04	262	-145.22	73	-150.60	482
		% Error	0.0067	0	0	0.38	0.0069	0	-0.0266	-0.415
	5	Predicted	-148.50	1	-157.53	326	-145.25	73	-151.39	567
		Exact	-148.49	1	-157.53	326	-145.24	73	-151.41	568
		% Error	0.0067	0	0	0	0.0069	0	-0.0132	-0.176
	6	Predicted	-148.50	1	-157.95	389	-145.26	73	-152.11	648
		Exact	-148.49	1	-157.96	390	-145.25	73	-152.11	647
		% Error	0.0067	0	-0.0063	-0.26	0.0069	0	0	0.155
	7	Predicted	-148.49	1	-158.33	453	-145.25	73	-152.74	725
		Exact	-148.49	1	-158.35	454	-145.25	73	-152.73	726
		% Error	0	0	-0.0126	-0.22	0	0	0.0065	-0.138
	8	Predicted	-148.49	1	-158.69	517	-145.26	73	-153.30	799
		Exact	-148.49	1	-158.69	517	-145.27	73	-153.27	799
		% Error	0	0	0	0	-0.0069	0	0.0196	0
	9	Predicted	-148.49	1	-159.02	581	-145.27	73	-153.75	872
		Exact	-148.49	1	-159.00	581	-145.27	73	-153.75	871
		% Error	0	0	0.0126	0	0	0	0	0.115
	10	Predicted	-148.49	1	-159.35	644	-145.27	73	-154.13	942
		Exact	-148.49	1	-159.29	644	-145.28	73	-154.18	942
		% Error	0	0	0.0376	0	-0.0069	0	-0.0324	0
+60	2	Predicted	-148.49	1	-159.20	135	-145.04	73	-149.94	285
		Exact	-148.49	1	-159.26	136	-145.05	73	-149.94	286
		% Error	0	0	-0.0377	-0.74	-0.0069	0	0	-0.35
	3	Predicted	-148.50	1	-160.45	199	-145.14	73	-151.75	383
		Exact	-148.49	1	-160.43	199	-145.14	73	-151.77	383
		% Error	0.0067	0	0.0125	0	0	0	-0.0132	0
	4	Predicted	-148.50	1	-161.34	263	-145.18	73	-153.22	479
		Exact	-148.49	1	-161.32	262	-145.18	73	-153.24	482
		% Error	0.0067	0	0.0124	0.38	0	0	-0.0130	-0.622
	5	Predicted	-148.50	1	-162.08	326	-145.21	73	-154.48	567
		Exact	-148.49	1	-162.06	326	-145.21	73	-154.46	568
		% Error	0.0067	0	0.0123	0	0	0	0.0129	-0.176
	6	Predicted	-148.49	1	-162.72	389	-145.22	73	-155.55	648
		Exact	-148.49	1	-162.70	390	-145.22	73	-155.51	647
		% Error	0	0	0.0123	-0.26	0	0	0.0257	0.154
	7	Predicted	-148.48	1	-163.26	454	-145.23	73	-156.45	726
		Exact	-148.49	1	-163.28	454	-145.24	73	-156.44	726

		% Error	-0.0067	0	-0.0122	0	-0.0069	0	0.0064	0
	8	Predicted	-148.47	1	-163.78	517	-145.25	73	-157.25	798
		Exact	-148.49	1	-163.79	517	-145.24	73	-157.25	799
		% Error	-0.0135	0	-0.0061	0	0.0069	0	0	-0.125
	9	Predicted	-148.48	1	-164.23	582	-145.25	73	-157.86	871
		Exact	-148.49	1	-164.26	581	-145.25	73	-157.97	871
		% Error	-0.0067	0	-0.0183	0.17	0	0	-0.0696	0
	10	Predicted	-148.47	1	-164.67	645	-145.26	73	-158.35	941
		Exact	-148.49	1	-164.69	644	-145.26	73	-158.26	942
		% Error	-0.0135	0	-0.0121	0.16	0	0	0.0568	-0.106
-20	2	Predicted	-144.86	136	-148.50	1	-143.75	286	-145.41	73
		Exact	-144.89	136	-148.49	1	-143.77	286	-145.40	73
		% Error	-0.0207	0	0.0067	0	-0.0139	0	0.0069	0
	3	Predicted	-144.50	199	-148.49	1	-143.18	385	-145.38	73
		Exact	-144.50	199	-148.49	1	-143.16	383	-145.37	73
		% Error	0	0	0	0	-0.0140	0.52	0.0069	0
	4	Predicted	-144.22	262	-148.49	1	-142.68	479	-145.36	73
		Exact	-144.21	262	-148.49	1	-142.67	482	-145.35	73
		% Error	0.0069	0	0	0	0.0070	-0.62	0.0069	0
	5	Predicted	-143.96	325	-148.48	1	-142.24	567	-145.34	73
		Exact	-143.96	326	-148.49	1	-142.26	568	-145.35	73
		% Error	0	-0.31	-00067	0	-0.0140	-0.18	-0.0069	0
	6	Predicted	-143.75	390	-148.49	1	-141.34	647	-145.34	73
		Exact	-143.75	390	-148.49	1	-141.34	647	-145.34	73
		% Error	0	0	0	0	0	0	0	0
	7	Predicted	-143.52	454	-148.49	1	-141.46	725	-145.35	73
		Exact	-143.52	454	-148.49	1	-141.45	726	-145.33	73
		% Error	0	0	0	0	0.0071	-0.14	0.0138	0
	8	Predicted	-143.38	518	-148.48	1	-141.33	799	-145.32	73
		Exact	-143.38	517	-148.49	1	-141.33	799	-145.33	73
		% Error	0	0.193	-0.0067	0	0	0	-0.0069	0
	9	Predicted	-143.23	582	-148.49	1	-141.09	871	-145.32	73
		Exact	-143.23	581	-148.49	1	-141.09	871	-145.33	73
		% Error	0	0.172	0	0	0	0	-0.0068	0
	10	Predicted	-143.08	643	-148.50	1	-140.88	941	-145.33	73
		Exact	-143.08	644	-148.49	1	-140.88	942	-145.33	73
		% Error	0	-0.16	0.0067	0	0	-0.11	0	0
-40	2	Predicted	-141.32	135	-148.49	1	-142.22	286	-145.50	73
		Exact	-141.30	136	-148.49	1	-142.23	286	-145.48	73
		% Error	0.0142	-0.74	0	0	-0.0070	0	0.0137	0
	3	Predicted	-140.51	199	-148.49	1	-141.02	384	-145.41	73
		Exact	-140.52	199	-148.49	1	-141.01	383	-145.43	73
		% Error	-0.0071	0	0	0	0.0071	0.26	-0.0138	0
	4	Predicted	-139.93	263	-148.49	1	-140.01	480	-145.38	73
		Exact	-139.93	262	-148.49	1	-140.02	482	-145.40	73

		% Error	0	0.38	0	0	-0.0071	-0.41	-0.0138	0
	5	Predicted	-139.44	326	-148.49	1	-139.21	567	-145.37	73
		Exact	-139.44	326	-148.49	1	-139.21	568	-145.38	73
		% Error	0	0	0	0	0	-0.17	-0.0069	0
	6	Predicted	-139.01	389	-148.49	1	-138.57	646	-145.37	73
		Exact	-139.01	390	-148.49	1	-138.57	647	-145.37	73
		% Error	0	-0.26	0	0	0	-0.15	0	0
	7	Predicted	-138.59	455	-148.49	1	-137.89	724	-145.38	73
		Exact	-138.60	454	-148.49	1	-137.89	726	-145.36	73
		% Error	-0.0072	0.22	0	0	0	-0.28	0.0137	0
	8	Predicted	-138.28	517	-148.49	1	-137.35	799	-145.35	73
		Exact	-138.28	517	-148.49	1	-137.35	799	-145.35	73
		% Error	0	0	0	0	0	0	0	0
	9	Predicted	-137.97	583	-148.48	1	-136.88	874	-145.36	73
		Exact	-137.97	581	-148.49	1	-136.87	871	-145.35	73
		% Error	0	0.344	-0.0067	0	0.0073	0.34	0.0069	0
	10	Predicted	-137.68	642	-148.50	1	-136.44	940	-145.34	73
		Exact	-137.68	644	-148.49	1	-136.44	942	-145.35	73
		% Error	0	-0.31	0.0067	0	0	-0.21	-0.0069	0

Table A.2 Simulation results for the test i.e patterns not seen during training (over training case)

Temp Change (degC)	Tran Time (min)		Inner radius				Outer radius			
			Max Stress		Min stress		Max Stress		Min stress	
			Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)	Value (10^8Pa)	Time (s)
+30	3	Predicted	-142.22	1	-155.34	427	-138.64	1011	-143.79	752
		Exact	-148.49	1	-154.46	199	-145.22	73	-148.54	382
		%Error	-4.2225	0	0.57	114.573	-4.531	1284.93	-3.198	96.859
	4	Predicted	-143.48	1	-155.26	437	-139.74	900	-144.63	1002
		Exact	-148.49	1	-154.90	262	-145.25	73	-149.28	481
		%Error	-3.374	0	0.232	66.794	-3.793	1132.877	-3.115	108.316
	5	Predicted	-144.50	1	-155.20	457	-140.72	801	-145.46	1217
		Exact	-148.49	1	-155.27	326	-145.26	73	-149.89	566
		%Error	-2.687	0	-0.045	40.184	-3.125	997.260	-2.955	115.017
	6	Predicted	-145.31	1	-155.17	488	-141.54	719	-146.22	1412
		Exact	-148.49	1	-155.90	391	-145.27	73	-150.41	646
		%Error	-2.142	0	-0.468	24.81	-2.568	884.932	-2.785	118.576
+50	3	Predicted	-150.14	100	-158.94	90	-147.38	186	-154.03	1030
		Exact	-148.49	1	-158.44	199	-145.17	73	-150.69	383
		%Error	1.111	9900.0	0.316	-54.774	1.522	154.794	2.216	168.929
	4	Predicted	-149.80	1	-159.96	178	-147.00	226	-155.23	1223
		Exact	-148.49	1	-159.18	262	-145.20	73	-151.92	482

		%Error	0.8822	0	0.490	-32.061	1.239	209.589	2.179	153.734
	5	Predicted	-149.54	1	-160.79	261	-146.70	257	-156.24	1389
		Exact	-148.49	1	-159.80	326	-145.22	73	-152.93	567
		%Error	0.7071	0	0.62	-19.939	1.019	252.055	2.164	144.973
	6	Predicted	-149.33	1	-161.48	340	-146.46	281	-157.08	1552
		Exact	-148.49	1	-160.33	391	-145.24	73	-153.81	647
		%Error	0.5657	0	0.717	-13.043	0.839	284.932	2.126	139.876
	3	Predicted	-131.54	1	-190.71	845	-133.68	73	-148.83	1579
		Exact	-148.49	1	-162.42	199	-145.11	73	-152.84	383
		%Error	-11.415	0	17.418	324.623	-7.877	0	-2.624	312.272
	4	Predicted	-133.48	1	-187.32	831	-133.99	1476	-150.44	1640
		Exact	-148.49	1	-163.46	262	-145.16	73	-154.56	482
+70		%Error	-10.108	0	14.597	217.176	-7.695	1921.918	-2.666	240.249
	5	Predicted	-135.60	1	-183.68	811	-134.55	1414	-151.90	1701
		Exact	-148.49	1	-164.32	326	-145.16	73	-155.98	567
		%Error	-8.681	0	11.782	148.773	-7.309	1836.986	-2.616	200.00
	6	Predicted	-137.70	1	-180.23	788	-135.40	1324	-153.17	1771
		Exact	-148.49	1	-165.07	391	-145.21	73	-157.22	647
		%Error	-7.266	0	9.184	101.535	-6.756	1713.699	-2.576	173.725
	3	Predicted	-171.74	1	-144.92	760	-162.87	73	-180.61	1531
		Exact	-142.51	199	-148.49	1	-142.08	383	-145.40	73
		%Error	20.51	-99.50	-2.404	75900	14.633	-80.94	24.216	1997.26
	4	Predicted	-171.77	1	-144.88	763	-162.87	73	-180.81	1531
		Exact	-142.07	262	-148.49	1	-141.35	482	-145.38	73
-30		%Error	20.91	-99.62	-2.431	76200	15.224	-84.86	24.371	1997.26
	5	Predicted	-171.10	1	-144.92	758	-162.83	73	-180.46	1530
		Exact	-141.70	326	-148.49	1	-140.74	567	-145.36	73
		%Error	20.75	-99.69	-2.404	75700	15.696	-87.125	24.147	1995.89
	6	Predicted	-169.56	1	-145.05	743	-162.72	73	-179.35	1529
		Exact	-141.38	391	-148.49	1	-140.21	647	-145.35	73
		%Error	19.93	-99.74	-2.317	74200	16.054	-88.72	23.392	1994.52
	3	Predicted	-127.98	853	-161.52	1	-133.49	1844	-141.03	73
		Exact	-138.53	199	-148.49	1	-139.93	383	-145.45	73
		%Error	-7.62	328.64	8.775	0	-4.602	381.46	-3.039	0
	4	Predicted	-127.93	854	-161.77	1	-133.49	1943	-141.02	73
		Exact	-137.79	262	-148.49	1	-138.70	482	-145.42	73
		%Error	-7.16	225.95	8.943	0	-3.756	303.112	-3.026	0
-50	5	Predicted	-127.92	854	-161.58	1	-133.49	2028	-141.02	73
		Exact	-137.17	326	-148.49	1	-137.69	567	-145.40	73
		%Error	-6.74	161.96	8.815	0	-3.050	257.67	-3.012	0
	6	Predicted	-127.94	855	-161.00	1	-133.49	2108	-141.03	73
		Exact	-136.64	390	-148.49	1	-136.81	647	-145.38	73
		%Error	-6.36	119.23	8.425	0	-2.427	225.81	-2.992	0

Table A.3 Simulation results for the test patterns i.e. patterns not seen during training (adequate training case)

Temp Change (degC)	Tran Time (min)		Inner radius				Outer radius			
			Max Stress		Min stress		Max Stress		Min stress	
			Value (MPa)	Time (s)	Value (MPa)	Time (s)	Value (MPa)	Time (s)	Value (MPa)	Time (s)
+30	3	Predicted	-148.45	1	-153.16	221	-145.56	73	-148.07	424
		Exact	-148.49	1	-154.46	199	-145.22	73	-148.54	382
		%Error	-0.02	0	-0.84	11.05	0.23	0	-0.31	10.99
	4	Predicted	-148.62	1	-153.62	292	-145.73	73	-148.70	517
		Exact	-148.49	1	-154.90	262	-145.25	73	-149.28	481
		%Error	0.08	0	-0.82	11.45	0.33	0	-0.38	7.48
	5	Predicted	-148.97	1	-153.56	348	-145.61	73	-148.76	582
		Exact	-148.49	1	-155.27	326	-145.26	73	-149.89	566
		%Error	0.32	0	-1.10	6.74	0.24	0	-0.75	2.82
	6	Predicted	-148.87	1	-153.74	408	-145.31	73	-149.02	670
		Exact	-148.49	1	-155.90	391	-145.27	73	-150.41	646
		%Error	0.25	0	-1.38	4.34	0.02	0	-0.92	3.71
+50	3	Predicted	-148.41	1	-159.37	173	-144.96	73	-151.30	352
		Exact	-148.49	1	-158.44	199	-145.17	73	-150.69	383
		%Error	-0.05	0	0.58	-13.1	-0.14	0	0.40	-8.09
	4	Predicted	-148.27	1	-160.67	259	-145.14	73	-152.97	480
		Exact	-148.49	1	-159.18	262	-145.20	73	-151.92	482
		%Error	-0.14	0	0.93	-1.14	-0.04	0	0.69	-0.41
	5	Predicted	-148.33	1	-161.15	303	-145.06	73	-153.68	532
		Exact	-148.49	1	-159.80	326	-145.22	73	-152.93	567
		%Error	-0.10	0	0.84	-7.05	-0.11	0	0.49	-6.17
	6	Predicted	-148.33	1	-161.80	361	-145.04	73	-154.66	619
		Exact	-148.49	1	-160.33	391	-145.24	73	-153.81	647
		%Error	-0.10	0	0.91	-7.67	-0.13	0	0.55	-4.32
+70	3	Predicted	-148.37	1	-160.26	207	-145.16	73	-151.11	401
		Exact	-148.49	1	-162.42	199	-145.11	73	-152.84	383
		%Error	-0.08	0	-1.32	4.02	0.03	0	-1.13	-4.69
	4	Predicted	-148.28	1	-161.75	274	-145.31	73	-153.55	503
		Exact	-148.49	1	-163.46	262	-145.16	73	-154.56	482
		%Error	-0.14	0	-1.06	4.58	0.10	0	-0.65	4.35
	5	Predicted	-148.83	1	-162.29	324	-145.84	73	-154.98	564
		Exact	-148.49	1	-164.32	326	-145.16	73	-155.98	567
		%Error	0.22	0	-1.23	-0.61	0.46	0	-0.64	-0.53
	6	Predicted	-148.79	1	-162.72	383	-145.46	73	-155.93	630
		Exact	-148.49	1	-165.07	391	-145.21	73	-157.22	647
		%Error	0.20	0	-1.42	-2.04	0.17	0	-0.82	-2.62
-30	3	Predicted	-140.45	228	-148.30	1	-140.41	427	-143.86	73
		Exact	-142.51	199	-148.49	1	-142.08	383	-145.40	73
		%Error	-1.44	14.57	-0.12	0	-1.17	11.48	-1.05	0

	4	Predicted	-140.41	255	-148.36	1	-140.40	479	-145.30	73
		Exact	-142.07	262	-148.49	1	-141.35	482	-145.38	73
		%Error	-1.16	-2.67	-0.08	0	-0.67	-0.62	-0.05	0
	5	Predicted	-140.08	317	-148.35	1	-139.78	549	-145.49	73
		Exact	-141.70	326	-148.49	1	-140.74	567	-145.36	73
		%Error	-1.98	-2.76	-0.09	0	-0.68	-3.17	0.08	0
	6	Predicted	-139.37	381	-148.38	1	-139.07	628	-145.48	73
		Exact	-141.38	391	-148.49	1	-140.21	647	-145.35	73
		%Error	-1.80	-2.55	-0.07	0	-0.81	-2.93	0.08	0
-50	3	Predicted	-141.03	178	-148.17	1	-141.55	353	-141.02	73
		Exact	-138.53	199	-148.49	1	-139.93	383	-145.45	73
		%Error	1.80	-10.6	-0.21	0	-1.15	-7.83	-3.04	0
	4	Predicted	-139.81	248	-148.30	1	-140.02	459	-142.19	73
		Exact	-137.79	262	-148.49	1	-138.70	482	-145.42	73
		%Error	1.46	-5.34	-0.12	0	0.95	-4.77	-2.22	0
	5	Predicted	-139.01	316	-148.35	1	-139.08	545	-144.80	73
		Exact	-137.17	326	-148.49	1	-137.69	567	-145.40	73
		%Error	1.34	-3.06	-0.09	0	1.00	-3.88	-0.41	0
	6	Predicted	-138.23	379	-148.42	1	-138.43	625	-145.41	73
		Exact	-136.64	390	-148.49	1	-136.81	647	-145.38	73
		%Error	1.16	-2.82	-0.04	0	1.18	-3.40	0.02	0

Bibliography

[1] Bathe K. J. "Finite Element Procedure" (Prentice Hall of India Private Limited, First Edition, 1996).

[2] Chandrupatla T.R, Belegundu A.D. "Introduction To Finite Elements in Engineering" (Prentice Hall of India Private Limited, Third Reprint, 1997).

[3] Carl C.Osgood "Fatigue Design" (The Pergamon Testbook Inspection Copy Service, Second Edition)

[4] D. T. Lin, P. A. Ligomenides and J. E. Dayhoff "Learning with the Adaptive Time-Delay Neural Network" I. S.R. T.R.93-49

[5] Heykin Symon "Neural Networks" MackMillan, Newyork,1994

[6] N. K. Mukhopadhyay, B. K. Dutta, H. S. Kushwaha "Generation of Data Base for On-Line Fatigue Life Monitoring of Indian Nuclear Power Plant Components" BARC Internal Report 1994.

[7] N. K. Mukhopadhyay, B. K. Dutta, H. S. Kushwaha, P. SwamiPrasad, A. Kakodkar "Implementation of Finite Element Based Fatigue Monitoring System at Heavy Water Plant Kota" Nuclear Engineering and Design, Elsevier 187 (1999), 153-159

[8] Paul J. Werbose "Backpropagation Through Time : What It Does and How To Do It" Proceedings of IEEE, Vol 78, No.10, Oct 1990, pp 1550-1559.

[9] Reddy J.N. "Introduction to Finite Element Method" McGraw Hill, 1998.

[10] R. Masini, E. Padovani, M.E. Ricotti, E. Zio "Dynamic Simulation of a Steam Generator by Neural Network", Nuclear Engineering and Design, Elsevier, 187 (1999), pp 197-213.

[11] Shawn P. D., Michael R. D., " Continuous Time Temporal Back-Propagation with Adaptable Time Delays" IEEE Transactions on Neural Networks, Vol 4, No.2, March 1993, pp 328-339.

[12] Zurada J.M. "Introduction to Artificial Neural Networks", Jaico Publishing House 1997.